UCRL-MA-131161 Rev. 1

SafePatch

(formerly Secure Software Distribution System (SSDS))

Version 0.9

User Manual

March, 1999

99.096

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or the University of California, and shall not be used for advertising or product endorsement purposes.

CIAC is the U.S. Department of Energy's Computer Incident Advisory Capability. Established in 1989, shortly after the Internet Worm, CIAC provides various computer security services to employees and contractors of the DOE, such as:

- Incident Handling Consulting
- Computer Security Information
- On-site Workshops
- White-hat Audits

CIAC is located at Lawrence Livermore National Laboratory and is a part of its Computer Security Technology Center. CIAC is also a founding member of FIRST, the Forum of Incident Response and Security Teams, a global organization established to foster cooperation and coordination among computer security teams worldwide.

Reference to any specific commercial product does not necessarily constitute or imply its endorsement, recommendation or favoring by CIAC, the University of California, the United States Department of Energy, or the United States Government.

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.

Work performed under the auspices of the U. S. Department of Energy by Lawrence Livermore National Laboratory under Contract W-7405-Eng-48.

TABLE OF CONTENTS

1.0 What is SafePatch?

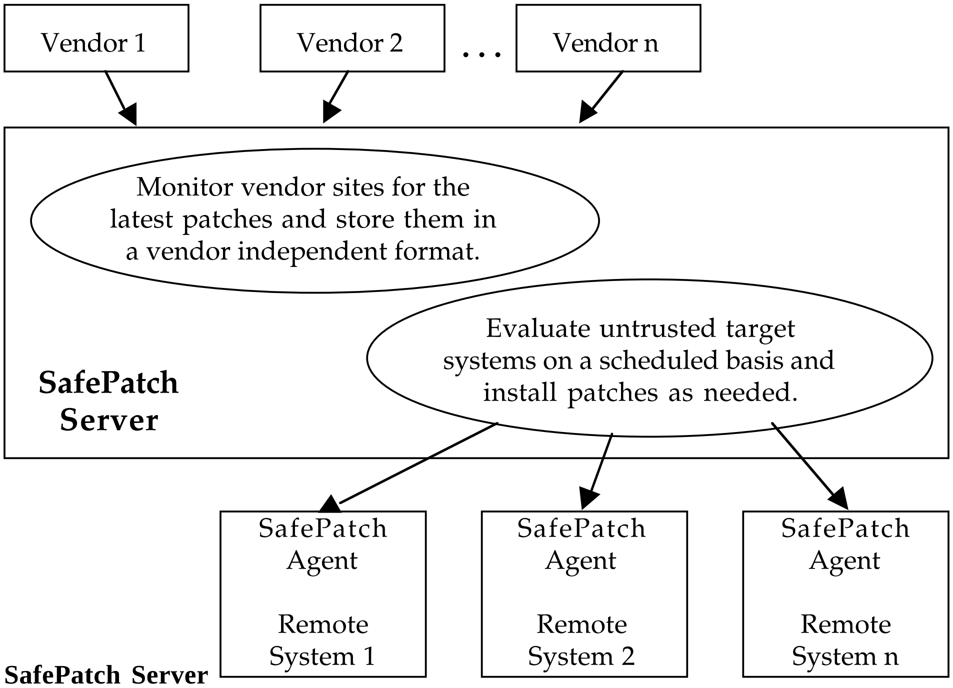
The SafePatch version 0.9 provides automated analysis of network-based computer systems to determine the status of security patches and distributes needed patches. SafePatch determines what patches need to be installed and what patches are installed on a system. SafePatch will distribute needed patches to the remote system for later installation. For those patches that are installed, SafePatch checks the permissions and ownership of the files referenced in the patch and reports on the attributes that differ from those recommended by the patch.

SafePatch also ensures that the system software is authentic (that is, belonging to either a release of an operating system or a patch). The process SafePatch uses to authenticate the software on a system is more reliable and secure than other vendor-specific tools. SafePatch compares the remote system's files with the files from the patches to determine what is actually installed and what needs to be installed. This approach ensures accurate reporting of a system's patch status. It also allows SafePatch to identify files that do not belong to either the original system distribution (for example, Solaris 2.5) or to any patch. These unidentified files may be customized or trojan. Either way these files should be investigated further to determine their exact origin.

SafePatch version 0.9 is supported on SUN systems running SunOS 5.5.1 or Solaris 2.5.1 and higher.

1.1 SafePatch Overview

SafePatch has two major components: SafePatch Server and SafePatch Agent. This section describes the roles of each component.


Vendor 1
Vendor 2
. . .
Vendor n
Monitor vendor sites for the latest patches and store them in a vendor independent format.
Evaluate untrusted target systems on a scheduled basis and install patches as needed.
SafePatch Server
SafePatch Agent
Remote System 1
SafePatch Agent
Remote System 2
SafePatch Agent
Remote System n

1.1.1 SafePatch Server

The SafePatch Server is the centralized server from which the analysis of remote systems can be scheduled and monitored. The SafePatch Server has two major tasks:

1. Monitoring and collecting patches from vendors' ftp sites.
2. Evaluating remote systems and distribution of patches.

1.1.1.1 Monitoring and Patch Collecting from Vendors' FTP Sites

To determine what patches need to be installed on a system, a software management tool such as SafePatch needs to know when new patches are available. The SafePatch Server accomplishes this by monitoring vendors' ftp sites on a regular basis and pulling across new or upgraded patches. The SafePatch administrator can specify which vendor sites to monitor and which patches to collect. For instance, SafePatch can be configured to collect only Solaris 2.5 security patches. The collected patches are then converted to a standard patch format and stored in a patch database.

The Vendor Server is the part of the SafePatch Server responsible for monitoring ftp sites, collecting new and updated patches, and converting patches to the standard patch format.

1.1.1.2 Evaluating Remote Systems

Evaluating a remote system consists of comparing the system software with the patches in the patch database (collection of patches from the vendors' ftp sites converted to a standard patch format). The SafePatch Server accomplishes this by first querying the remote system for information about its operating system software and architecture. The SafePatch Server then retrieves all patches from the patch database pertaining to the remote system. The checksum, permissions, and owner settings on all the files and directories in these patches are compared with the files and directories on the remote system. The results of this evaluation indicate which patches are installed and which patches need to be installed on the remote system. The SafePatch Server also permits network administrators (and users) to ensure the integrity of the system software. For example, if the checksum of a file on the remote system does not match any of the files in the patch database, then the file could be trojan or customized. Either way, further investigation is needed to ensure that the system has not been compromised.

At the end of the evaluation process, needed patches can be distributed to the remote system for later installation. This permits local system administrators to install patches when it is most convenient for the system's users.

A report is also generated at the end of the evaluation process summarizing the findings. These findings include which patches have been evaluated, which patches are installed, which patches need to be installed, and what files were not identifiable.

The SafePatch administrator controls the evaluation of remote systems through the part of the SafePatch Server called the Patch Server. The evaluation of one or more systems is referred to as a job within the Patch Server. The time, date, and how often a job is to occur can be specified for each remote system or for a group of systems. The Patch Server interface also permits easy tracking of the status of an evaluation job and viewing of reports and logs.

1.1.2 SafePatch Agent

The SafePatch Agent must be installed on any system to be evaluated by SafePatch. The SafePatch Agent is a lightweight process that responds to the SafePatch Server's commands and requests. The Agent queries the system for information such as the checksum or the permissions of a file. This information is used by the Patch Server to determine which patches are installed on a system, and if any files on the system are unidentifiable.

1.2 How to Use This Manual

This manual, the *SafePatch User's Guide*, is a complete reference to SafePatch. It explains patch management concepts and provides step-by-step instructions for using the SafePatch Server software. The *SafePatch Installation Procedures* provides information on how to install and start the SafePatch Server and SafePatch Agent. The installation procedures are distributed with the SafePatch software.

Chapter 2, "Getting Started" describes how to log on and off the SafePatch Server. It also describes how to set and change the SafePatch password.

Chapter 3, "Vendor Server" describes the part of the SafePatch Server that manages the monitoring of ftp sites, collection of patches, and conversion of patches to the standard patch format. The first part of Chapter 3 describes the user interface and defines the data displayed in each form. The second part of Chapter 3 provides step-by-step instructions for configuring the Vendor Server. Instructions are provided for adding a new vendor site to be monitored, changing information about a site, deleting a site, and monitoring the patch collection process.

Chapter 4, "Patch Server" describes the part of the SafePatch Server that controls the evaluation of remote systems (also referred to as jobs) and the distribution of patches. Similar to Chapter 3, the first part of this chapter describes the Patch Server user interface and defines the data displayed in each form. The second part of the chapter provides step-by-step instructions for scheduling and monitoring the evaluation of remote systems and the distribution of patches. Instructions are provided for configuring the Patch Server, testing communications between the Patch Server and remote systems, scheduling jobs, and reviewing the reports and logs.

Chapter 5, "Interpreting Reports" gives details on the information provided in a report. The reports indicate which patches need to be installed and the order they should be installed. It also identifies files with the wrong permissions and access controls, unidentified files, and patches that are installed.

Chapter 6, "Interpreting Log Files" gives details on the messages provided in the log files. A log file is generated for each evaluation job scheduled using the Patch Server. This chapter is a good place to look to determine if there is a problem with a job.

Chapter 7, "Troubleshooting" provides solutions to common problems related to SafePatch.

1.3 What's New?

There are many new features and bug fixes in SafePatch version 0.9. The most significant enhancement is the ability to distribute patches to the remote system. Just select the download check box when scheduling a job. After the remote system is evaluated, all needed patches will be combined into a tar file and downloaded to the remote system.

To accommodate the distribution of patches, new fields were added to SafePatch's standard patch format. All SSDS 0.5 patches must be converted to this new format. A translator has been provided with SafePatch 0.9 to assist with this conversion.

To translate patches:

1. Move to the SafePatch utility directory.

   ```
   >cd  SafePatch/util
   ```

2. Run the translation script.

   ```
   > ./Translate_0.5_to_0.9.sh psDir trans.log
   ```

   where psDir is the directory where the SafePatch standard formatted patches are stored (*e.g.*, PSDB) and trans.log is a file reporting the status of the patch translation.

Other significant changes to SafePatch include:

1. The user interface has been modified to include:
   a. Table content aligned in columns and permitting moving of columns.
   b. Confirmation messages are displayed when the daemon processes are started and stopped, and ASAP jobs are scheduled.
   c. The Vendor Service Type field on the Vendor Server main window has been removed.
   d. ASAP jobs no longer require a start date when scheduled.

2. The conversion script for Sun Solaris patches provided with SafePatch 0.9 has been enhanced to increase the number of patches it successfully converts.

3. Job scheduling has been fixed to support scheduling of jobs at midnight, leap year, and scheduling jobs one or more years in advance.

4. Hosts added to the Host list are checked to ensure that they are alive and that the SafePatch Agent is running on the remote system. SafePatch 0.9 permits a host to be added even if the Agent is not running or the host is not alive. However, the user is warned in this case.

5. The host running the SafePatch Server is automatically added to the Host List at installation. This system cannot be deleted from the Host list.

6. A new report option has been added to the Vendor Server. The report provides status information on the patch collection and conversion process.

7. Java was upgraded to version 1.2 beta 6. SafePatch 0.9 will not work with any prior versions of Java.

8. If the patch conversion script doesn't exist in the SafePatch/binm/ES/smelt_scripts directory, a file selection window will be displayed. Once the script is located, it is moved (not copied) to the smelt_scripts directory.

With the move to Java 1.2 beta 6 we have identified a problem with the DSA digital signatures used to sign and verify the password. For this reason, passwords have been disabled. The login password will be enabled once Java has been fixed.

Please report any bugs to SafePatch@cheetah.llnl.gov.

1.4 Future Work

Version 0.9 of SafePatch addresses one of the most time consuming tasks of system administration, namely the determination of which patches to install. Version 0.9 of SafePatch provides the capability to distribute needed patches to remote systems. This permits local system administrators to install patches without interrupting the system's users. Another time consuming task is the actual installation of those patches. Future releases of SafePatch will address the automated installation of security patches as well as the ability to "back-out" installed patches, restoring a system's previous state.

Future versions of SafePatch will support other operating systems, such as HPUX, Digital Unix, and SGI.

1.5 Further Information

To obtain SafePatch, or information about it, contact:

Center for Information Operations and Assurance
Attention: SafePatch Project
Lawrence Livermore National Laboratory
P. O. Box 808, L-303
Livermore, CA 94551
- -
Telephone: (925) 423-6224
email: safepatch@cheetah.llnl.gov
www: http://ciac.llnl.gov/cstc/safepatch

2.0 Getting Started

This chapter provides instructions for a few basic commands, namely logging in and out of SafePatch, as well as setting up or changing the SafePatch password.

NOTE: With the move to Java 1.2 beta 6 we have identified a problem with the DSA digital signatures used to sign and verify the password. For this reason, passwords have been disabled. The login password will be enabled once Java has been fixed.

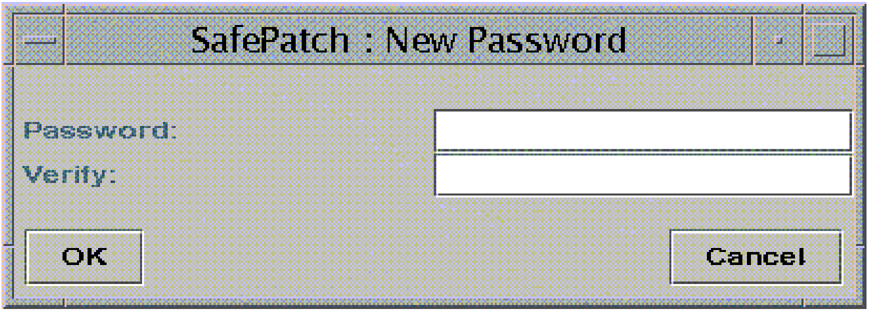
2.1 Logging into the SafePatch Server

To start the SafePatch Server:

1. Change directories to Safepatch-0.9/binm or set the PATH environment variable to include the SafePatch-0.9/binm directory. Type:

   ```
   >> SafePatch
   ```

2. For security reasons, a password is required to log into SafePatch. When SafePatch is logged into, a window will be displayed prompting for a new SafePatch password.

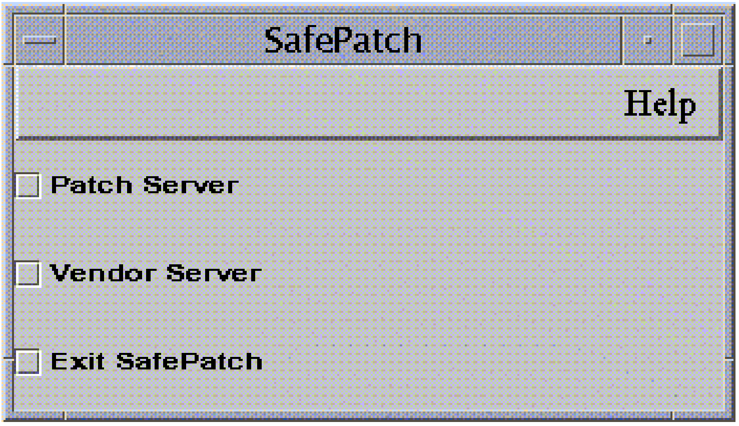
   Enter a new password in the **Password** and **Verify** fields**.** Passwords must be eight characters or longer.

   - or –

3. Select the **Cancel** button to abort the login process.

   - or –

   Select the **OK** button to log into SafePatch. The main SafePatch Server window will be displayed upon successful login. The Vendor Server and Patch Server can be accessed from this window.

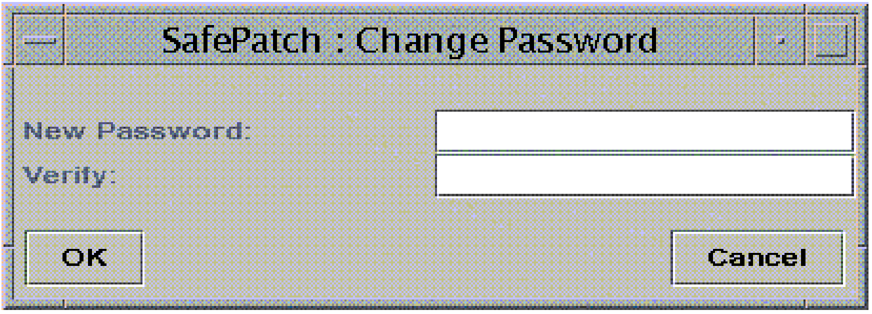


2.2 Changing the Password

The SafePatch password can be changed from both the Vendor Server and the Patch Server. To change the SafePatch password:

1. Select either **Vendor Server** or **Patch Server** from the main window.

2. Select **Change Password…** from the **File** menu. The Change Password window will be displayed.



3. Enter the **New Password**. Verify the new password by entering it again in the **Verify** field.

4. Select **OK** to change the password, or **Cancel** to reinstate the old password.

2.3 Logging Out of SafePatch

The **Exit SafePatch** option on the main SafePatch window will close the graphical user interface but leave the daemon processes running. The daemon processes control the scheduling and execution of jobs within SafePatch.

To exit SafePatch:

1. Select the **Exit SafePatch** checkbox from the main SafePatch Server window.

3.0 Vendor Server

One component of the SafePatch Server is the Vendor Server. The Vendor Server collects new and upgraded patches from vendors' ftp sites, converts the patches to a standard patch format, and stores them in a patch database.

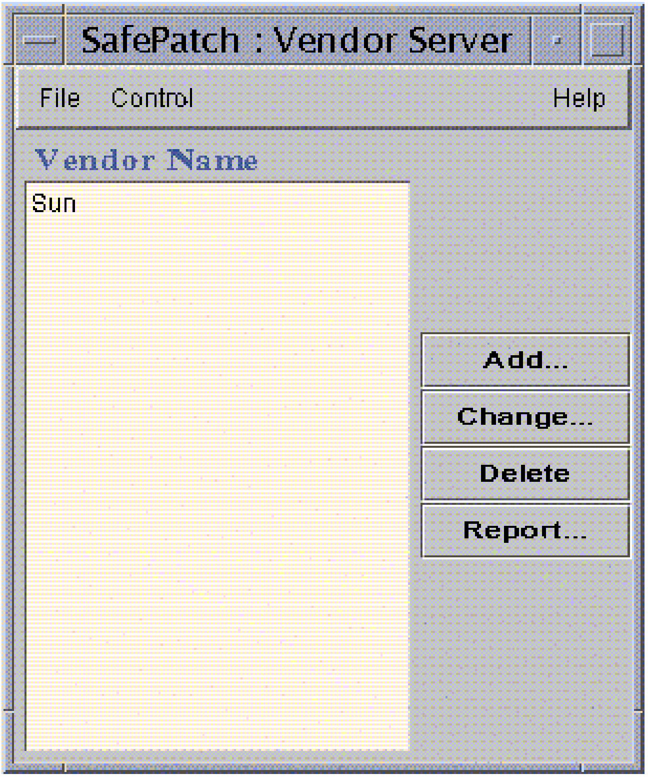
To configure the Vendor Server to collect patches from a vendor, information must be provided about a vendor's ftp site. Vendor information includes the site location, the user name/password to connect to the site, the directories where patches are stored, what files to collect, and how often to collect new and updated patches. It also includes the name of the script that converts vendor specific patches to a standard patch format.

Section 3.1, "Vendor Server Interface" describes the Vendor Server windows and the data required for each ftp site. Section 3.2, "How to Collect Patches from an ftp Site" provides instructions for adding a new vendor site, changing information about a vendor's site, deleting a vendor, and monitoring the patch collection process.

3.1 Vendor Server Interface

This section describes all of the Vendor Server windows, the data, and in each window the actions performed by each button.

3.1.1 Vendor Server Main Window



Sun

Vendor Name List containing all the ftp sites from which patches are being collected. Each site is assigned a unique vendor name.

Add… Add a site to be monitored. See section 3.1.2, “Add/Change Vendor Window.”

Change… Change information about a selected vendor. See section 3.1.2, “Add/Change Vendor Window.”

Delete Delete the selected vendor and stop collecting patches from this ftp site.

Report… View vendor patch collection and conversion results.

3.1.1.1 File Menu

Change Password… Allows the existing password to be changed. See section 2.2, “Changing the Password.”

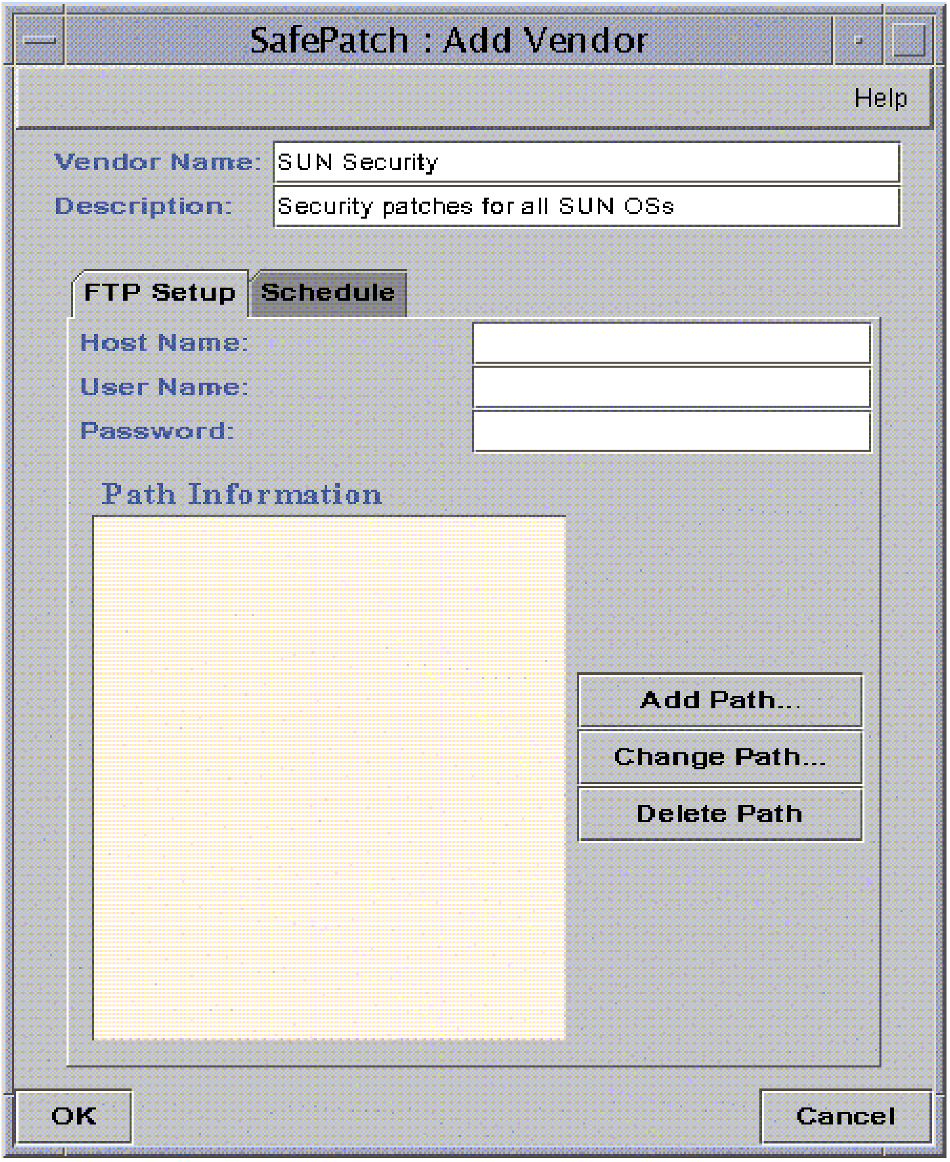
Exit Close the Vendor Server windows and return to the main SafePatch Server window.

3.1.1.2 Control Menu

The **Control** menu allows the SafePatch daemon processes to be started and stopped. The SafePatch daemon processes control the collection of patches from ftp sites as well as the scheduling and execution of jobs independent of the interface. Selecting the **Start Daemon** menu item will start these processes. The **Stop Daemon** menu item will stop these processes, abort all patch collections and jobs currently in process, and cause any pending patch collection and jobs not to run.

3.1.2 Add/Change Vendor Window

The only differences between the Add Vendor and Change Vendor windows are the window titles and which fields can be edited. All fields can be edited in the Add Vendor window. However, in the Change Vendor window, the **Vendor Name** field cannot be changed.



Vendor Name: SUN Security

Description: Security patches for all SUN OSs

Vendor Name A unique vendor name, such as Sun.

Description Short description of the types of patches collected from this vendor (*e.g.*, Solaris security patches for version 2.3 and higher).

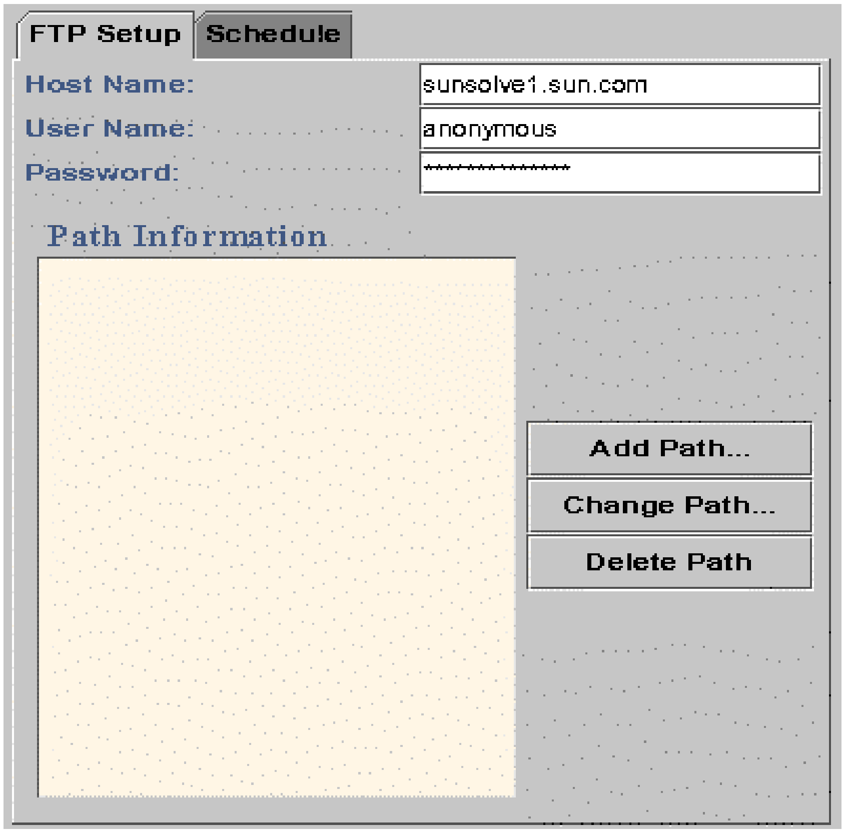
FTP Setup This folder contains information about the vendor's ftp site. See Section 3.1.2.1 for more information.

Schedule This folder contains information on how often to collect patches from the vendor's ftp site. See Section 3.1.2.2 for more information.

OK Applies and saves the data.

Cancel Return to the Vendor Server main window without saving changes.

3.1.2.1 Ftp Setup Folder



Host Name: sunsolve1.sun.com

User Name: anonymous

Password: ***************

This folder contains information on how to access a vendor's site and from which directories to collect patches.

Host Name The name of the vendor's ftp site.

User Name The user name used to log into the ftp site.

Password Password for the user name. Currently the password is saved in cleartext.

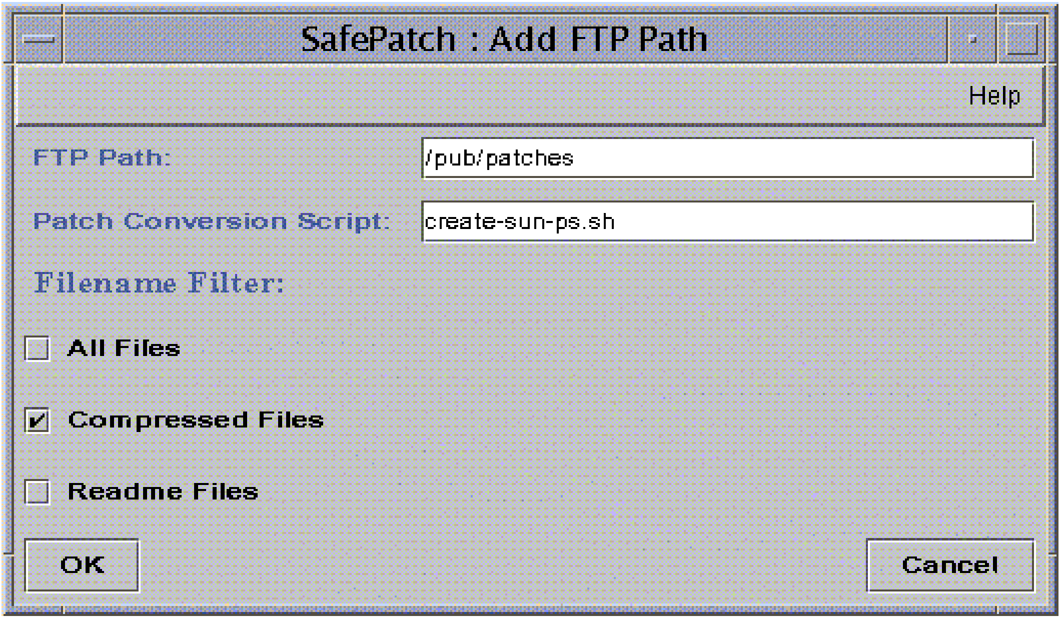
Path Information Directories containing patches to be monitored by SafePatch. Directories (paths) can be added to the list by selecting the **Add Path…** button.

Add Path… Add directories (paths) to the list. When selected, this button displays the Add Ftp Path window (see Section 3.1.2.1.1).

Change Path… Change information about an existing directory. When selected, this button displays the Change Ftp Path window (see Section 3.1.2.1.1).

Delete Path Delete a directory.

3.1.2.1.1 Ftp Path Window



FTP Path: /pub/patches

Patch Conversion Script: create-sun-ps.sh

Filename Filter:

□ All Files

☑ Compressed Files

□ Readme Files

FTP Path The full pathname to the directory where the patches reside.

Patch Conversion Script The name of the script converting the vendor's patches to the SafePatch standard patch format. create_sun_ps.sh is the script for converting Sun patches. create_sun_ps.sh is provided with SafePatch. The patch conversion scripts will be placed in the SafePatch-0.9/binm/ES/smeltscript directory.

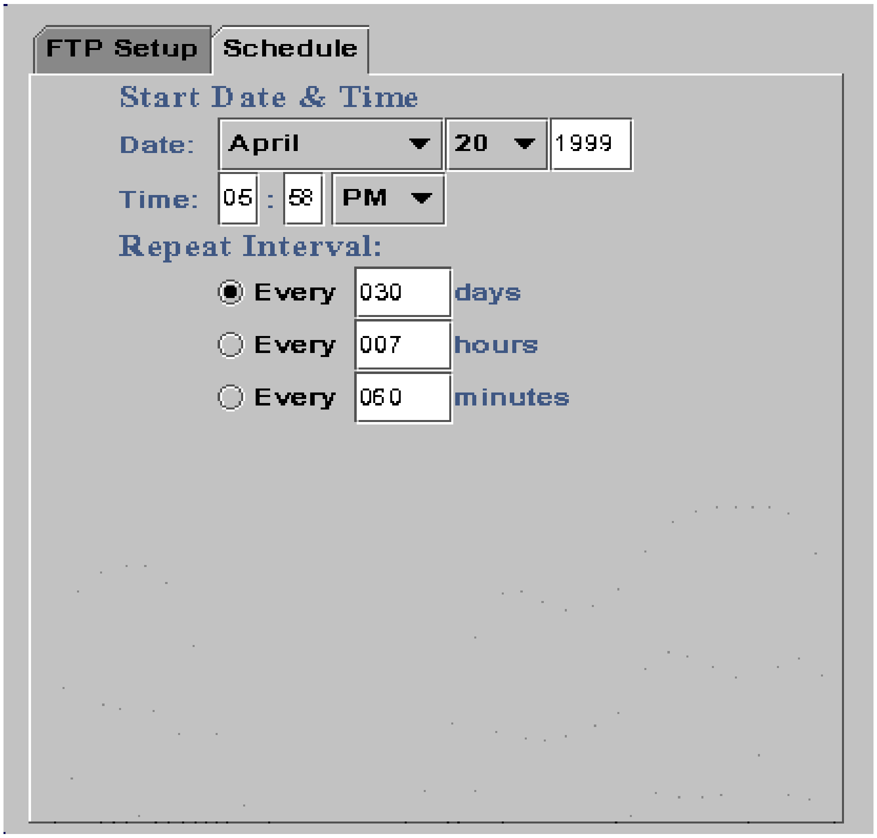
FileName Filter Three selections are available to filter information gathered from a directory: **All Files** collects any file in the directory, **Compressed Files** forces only files with a .tar.Z extension to be collected, and **Readme Files** forces only files with a .readme extension to be collected.

OK Saves data added or changed in the window.

Cancel Returns to the Add Vendor or Change Vendor window without saving changes.

3.1.2.2 Schedule Folder

This folder contains information on how often to monitor a vendor's ftp site for new and updated patches.



Start Date & Time

Date: April 20 1999

Time: 05 : 58 PM

Repeat Interval:

● Every 030 days

○ Every 007 hours

○ Every 060 minutes

Start Date & Time When to start collecting patches from a vendor’s site. By default, SafePatch fills in the current date and time.

Date The starting date in the form of “Month DD YYYY”

Time The time to start in the form of “HH:MM.” Select either “PM” or “AM.”

Repeat Interval How often to monitor the ftp site directories for new and updated patches. The frequency can be specified in days, hours, or minutes with the **Every ddd days**, **Every hhh hours**, and **Every mmm minutes** checkboxes.

3.1.3 Patch Collection and Conversion Window

This window provides information on the success of the collection of patches from a vendor’s ftp site and the conversion of those patches to the standard patch format.

Vendor Name Unique name associated with a vendor. A vendor can be listed multiple times: once for each path to a directory where patches reside and the last run time.

Ftp Information The name of the vendor’s ftp site.

Last Run Time The time when the patches were collected from the vendor’s ftp site and converted to the standard patch formats.

Collected If the vendor’s ftp site was accessed successfully, then this field lists the number of patches collected from the site (0 or more). Otherwise, this field will display “Failed” or “Interrupted.”

Converted If the patches collected were evaluated and converted successfully, then this field lists the number of patches converted. This number will be between “0” and the number of patches collected. If the conversion script could not be found or the conversion process was interrupted, then this field will list “Failed” or “Interrupted.” Selecting this field will display a list of patches converted.

Errors If the conversion script cannot process a patch, an error will be raised. The number of patches not successfully converted will be listed in this field. These patches must be evaluated and manually converted.

Selecting this field will display the list of patches that could not be converted successfully.

Delete Three log files are associated with each line in this table. To remove the log files and the table entry, select the Delete button.

OK Return to the main Vendor Server window.

3.2 How to Collect Patches from an ftp Site

This section provides step-by-step instructions on how to monitor ftp sites, collect new and updated patches, and monitor the patch collection process.

3.2.1 Adding a Vendor

To collect patches from a vendor not listed in the **Vendor Name** list in the main Vendor Server window:

1. Select the **Add…** button on the main Vendor Server window. The Add Vendor window will be displayed.

2. Fill in the **Vendor Name** and **Description** of the type of patches to collect.

3. In the **FTP Setup** folder, type in the **Host Name** of the ftp site (*e.g.*, Sun’s ftp server is sunsolve1.sun.com). Fill in the **User Name** and **Password** for logging into this server (*e.g.*, User Name = anonymous, Password = safepatch@llnl.gov).

   Note: Passwords are saved in cleartext.

4. Select the **Add Path…** button from the **FTP Setup** folder. The Add FTP Path window will be displayed.

5. In the Add FTP Path window, fill in the **FTP Path** field with the directory path where patches can be located. Fill in the name of the **Patch Conversion Script** to use to convert the vendor patches to the SafePatch standard patch format. create_sun_sp.sh is a script for Sun patches provided with SafePatch. This script converts the security patches for SunOS and Solaris versions 2.3 and higher. If the conversion script is not located in the directory SafePatch-0.9/binm/ES/smelt_scripts, a file chooser window will be displayed. Find and select the script, then select **Save**. The script will be moved to the smelt_scripts directory. Finally, select the types of files to collect from the specified ftp directory by selecting the appropriate buttons in the **Filename Filter** section. **All Files** will collect all the files found in the specified directory. **Compressed Files** will collect only files with a ".Z," .or ".tar.Z," extension. **Readme Files** will collect files with a “.readme” extension. Select the **OK** button to save this data and return to the **Add Vendor** window.

   To add multiple directories for this ftp site, select the **Add Path…** button again from the **Ftp Setup** folder, and repeat step 5 until all directories are added.

6. Select the **Schedule** folder on the Add Vendor window. Fill in the **Date** and **Time** to start collecting patches. By default, SafePatch enters the current date and time. Fill in the information

on how often to monitor the ftp site for new patches. For example, to monitor the ftp site once a week, select **Every ddd days** in the **Repeat Interval** section and enter 7 into the **ddd** textfield. To monitor the ftp site every 12 hours, select **Every hhh hours** in the **Repeat Interval** section and enter 12 into the **hhh** textfield.

7. When all the data in the **FTP Setup** and **Schedule** folders is entered, select **OK** to save the data and return to the main Vendor Server window. The Vendor will be added to the vendor list. Collecting patches from this site will be scheduled when the **OK** button is selected.

   - or -

   Select **Cancel** in the Add Vendor window to return to the main Vendor Server window without saving the data.

3.2.2 Changing Vendor Information

How often to check for new and updated patches, which directories to check, location and login information for a vendor's ftp site can be changed at any time. To change any of the vendor's ftp site information:

1. Select a vendor from the **Vendor Name** list in the main Vendor Server window.

2. Select the **Change…** button from the main Vendor Server window. The **Change Vendor** window will be displayed. The **Change Vendor** window is the same as the **Add Vendor** window (see details in section 3.2.1).

3. Modify the existing data in the form. Note: the **Vendor Name** cannot be changed.

4. Select the **OK** button to apply the changes. The **Cancel** button will return to the main Vendor Server window without saving the changes.

3.2.3 Deleting a Vendor

Patches will not be collected from a vendor's ftp site that is deleted from the vendor list in the main Vendor Server window. Deleting a vendor will not delete any patches collected previously for this vendor. To remove a vendor and all the vendor's associated data:

1. Select the vendor to be deleted from the **Vendor Name** list in the main Vendor Server window.

2. Select the **Delete** button from the main Vendor Server window. Deleting a vendor will stop SafePatch from monitoring and collecting patches from the vendor's ftp site. A Confirm Delete window will be displayed.

3. Select the **Yes** button to delete the vendor. Select the **No** button to keep the vendor and return to the main Vendor Server window.

3.2.4 Monitoring Patch Collection

The Report menu has been provided to assist with monitoring status of the patch collection and conversion processes.

1. Select the **Report…** button from the main Vendor Server window. The Patch Collection & Conversion Report window will be displayed (see section 3.1.3). This window displays the number of patches collected, converted, and failed conversion for each vendor's ftp directory.

   If the **Collected** cell contains "Failed," then the collection process was not started.

If the **Collected** cell contains “Interrupted,” then the collection process was started but never completed.

2. Select a **Converted** field. If the field contains a number, then the Patches Collected window will be displayed listing all patches successfully converted.

   If the **Converted** field contains “Failed,” then the conversion process was not started.

   If the **Converted** field contains “Interrupted,” then the conversion script was interrupted during its processing.

3. Select an **Error** field. If the field contains a number greater than zero, then the patches that could not be converted will be displayed. These patches must be evaluated and converted manually.

3.2.5 Deleting Patch Collection Log Files

The patch collection and conversion process creates three log files each time it is executed (i.e., three log files exist for each row in the Patch Collection & Conversion window). These log files should be periodically deleted. To delete the log files:

1. Select a vendor from the **Vendor Name** column on the Patch Collection & Conversion window.

2. Select the **Delete** button. A dialog box will be displayed to verify the delete.

3. Select the **Yes** button to delete the log file (and the row in the table).

   - or –

   Select **No** to abort the delete processes.

4.0 Patch Server

The Patch Server is the component of the SafePatch Server that controls the activities performed on remote systems. The activities performed on remote systems are referred to as jobs. For instance, the evaluation of patches on a remote system is referred to as a job.

The Patch Server is used to process and manipulate jobs throughout their lifecycle. Any activity that can be performed on a remote system is a job. A job can be processed on a single host or a group of hosts. Jobs can be created as continually scheduled processes, or one-time processes. They can be scheduled to start at a future date and time, or immediately. When a job has completed its lifecycle, a report will be generated which describes the results. The Patch Server interface consists of three tabbed folders: Scheduled, Pending, and Completed which enable the user to track jobs from initialization to completion.

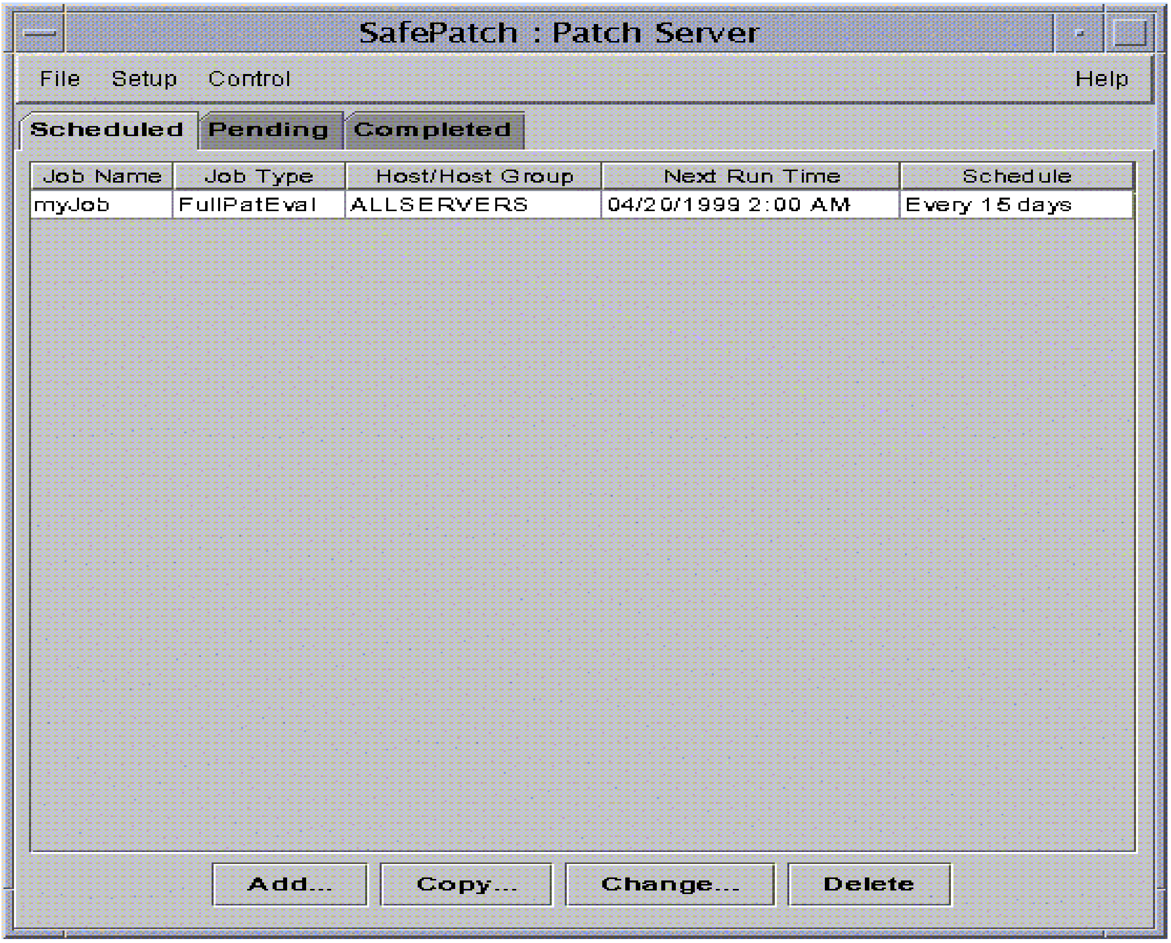
Section 4.1 “Patch Server Interface” describes the Patch Server windows and required data for scheduling and monitoring jobs. Section 4.2, “How to Configure the Patch Server” provides step-by-step instructions on adding and deleting remote systems to SafePatch, as well as grouping remote systems for faster and easier job scheduling. Section 4.3, “Testing Host Communications between the Patch Server and Remote Systems” provides instructions to determine if a SafePatch Agent is running on the remote system.

Section 4.4, “How to Schedule Jobs” provides step-by-step instructions for scheduling new jobs, changing, or deleting jobs. Sections 4.5 and 4.6 provide instructions on generating a report from the results of a job and saving the report. See Chapter 5.0 for a detailed description on the contents of reports. Finally, Section 4.7, “Viewing Log Files” provides instructions on how to display the log file for a job. The log file contains messages tracking the job through its lifecycle. See Chapter 6.0 for information on how to interpret the log file.

4.1 Patch Server Interface

This section describes all of the Patch Server windows, the data in each window, and the actions performed by each button.

4.1.1 Patch Server Main Window



Job Name	Job Type	Host/Host Group	Next Run Time	Schedule
myJob	FullPatEval	ALLSERVERS	04/20/1999 2:00 AM	Every 15 days

4.1.1.1 File Menu

The **File** menu on the main Patch Server window has two menu items:

Change Password… Changes the SafePatch password. See Section 2.2.

Exit Closes the Patch Server windows and returns to the main SafePatch Server window.

4.1.1.2 Setup Menu

Define Host List… Identifies hosts and configures groups of host. Selecting this menu item will display the Define Host List window (see Section 4.1.2).

4.1.1.3 Control Menu

Start Daemon Activates processes that control the scheduling and execution of jobs.

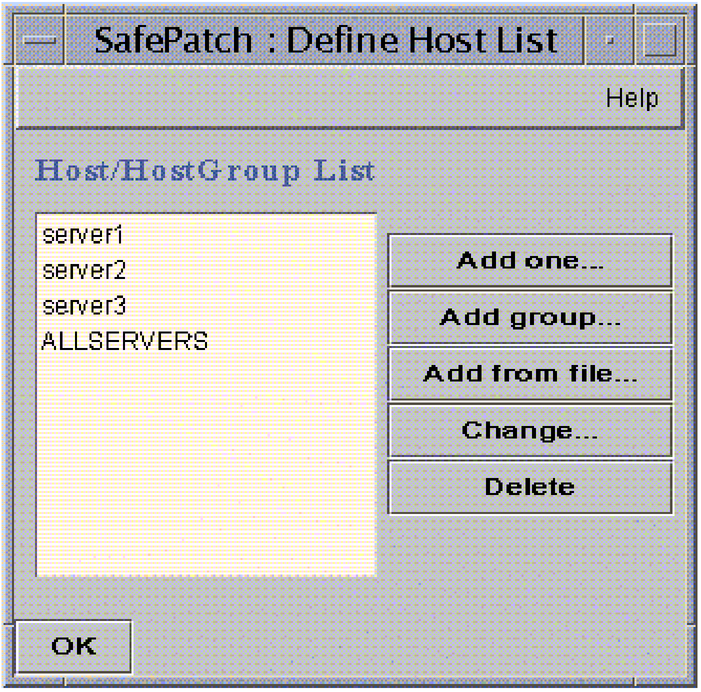
Stop Daemon Stops the processes controlling the scheduling and execution of jobs. Any pending jobs will not run.

Test Host

Communications… Tests communications between the remote host and the SafePatch Server.

4.1.2 Define Host List Window

Selecting the **Define Host List…** button from the **Setup** menu displays the Define Host List window and allows the setup of remote systems.



server1
server2
server3
ALLSERVERS

Host/Host Group List List of remote systems and host groups which can be evaluated by SafePatch. A host group is a group of one or more hosts permitting quick scheduling of jobs on multiple systems.

Add one… Add a remote system to the **Host/Host Group List**. When this button is selected, the Add Host window is displayed. See section 4.1.2.1 for more information on the Add Host window.

Add group… Displays the Add Host Group window which provides the capability to create a new host group and add it to the **Host/Host Group List**. See section 4.1.2.2 for details on the Add Host Group window.

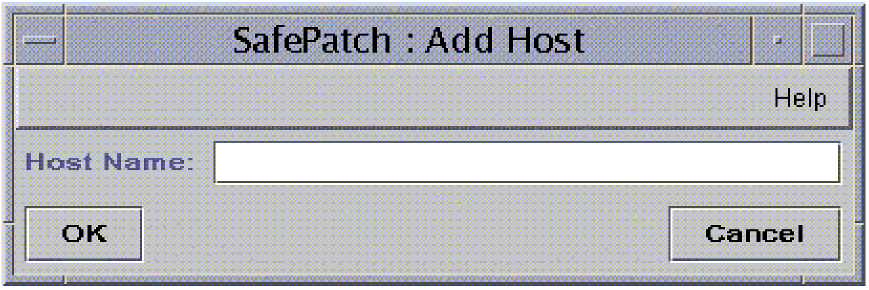
Add from file… Displays the Add From File window. Provides the capability to add a list of remote systems to the **Host/Host Group List** and to create a new host group by reading host names from a text file. See Section 4.1.2.3 for details on the Add From File window.

Change… Allows the host or host group information to be changed. The Change function is used to modify an individual host name, or the name of a host group as well as the host group members. See Section 4.1.2.4 for details on the change Host/Host Group window.

Delete Removes a selected host or host group from the **Host/Host Group List**.

4.1.2.1 Add Host Window

When the **Add one…** button is selected, the Add Host window is displayed.



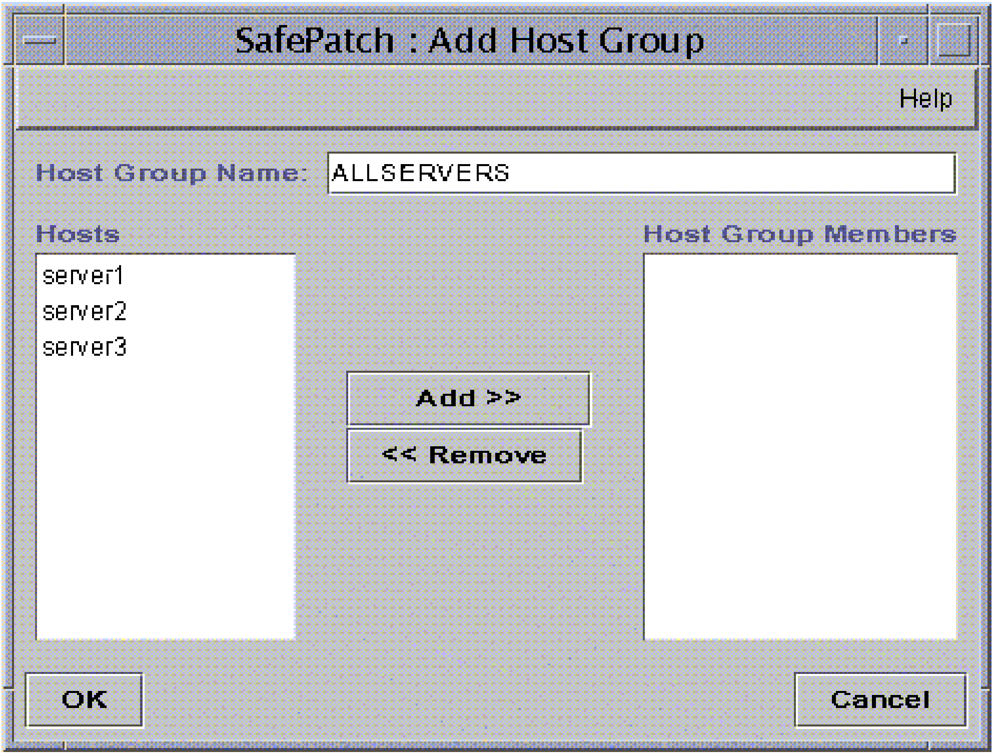
Host Name Name of the system being added to the host list.

OK Adds a host to the **Host/Host Group List**. If the SafePatch Agent software is not installed or running on the host, a message will be displayed warning that the host cannot be reached.

Cancel Returns to the main Patch Server window without adding the host to the list.

4.1.2.2 Add Host Group Window

Selecting the **Add group…** button displays the Add Host Group window. A host group is a conceptual grouping of systems to make it easier to schedule jobs on multiple systems simultaneously. For example, a group called SERVERS could be created which includes hosts Server1, Server2, and Server3. The individual hosts from the host list are displayed in the **Hosts** list to the left of the **Add Host Group** window. The **Host Group Members** list to the right of the **Add Host Group** window lists all hosts belonging to the host group. A host can belong to one or more host groups.



Host Group Name: ALLSERVERS

server1
server2
server3

Host Group Name Name of the new host group. The name must be unique and is forced to be capitalized.

Hosts List of all hosts not members of the host group.

Host Group Members List of the hosts that are members of the new host group.

Add >> Moves the selected hosts to the **Host Group Members** list.

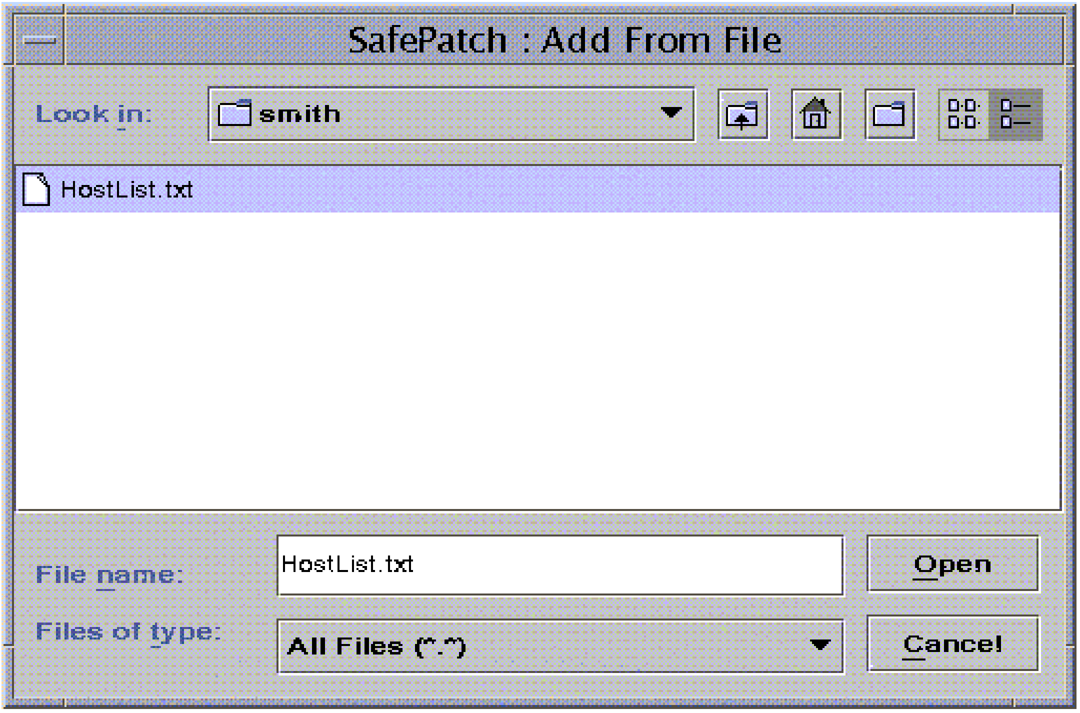
<< Remove Removes the selected hosts from the **Host Group Members** list.

OK Saves the new host group.

Cancel Returns to the Define Host List window without saving the host group.

4.1.2.3 Add From File Window

Selecting the **Add from file…** button displays the Add From File window. Host names read from a text file are added to the host list. This is an easy way to add multiple remote systems to the host list. It will also create a host group consisting of all the hosts in the file.


SafePatch : Add From File

Look in: smith

HostList.txt

File name: HostList.txt

Files of type: All Files (*.*)

Look in: Current directory. Selecting this button lists all the directories above the current directory (i.e., parents of the current directory).

File name: Name of file in specified directory containing a list of hosts.

Files of type: Filters filenames by file extension.

Open Reads the file and adds hosts from the specified file to the host list.

Cancel Returns to the Define Host List window without reading a file.

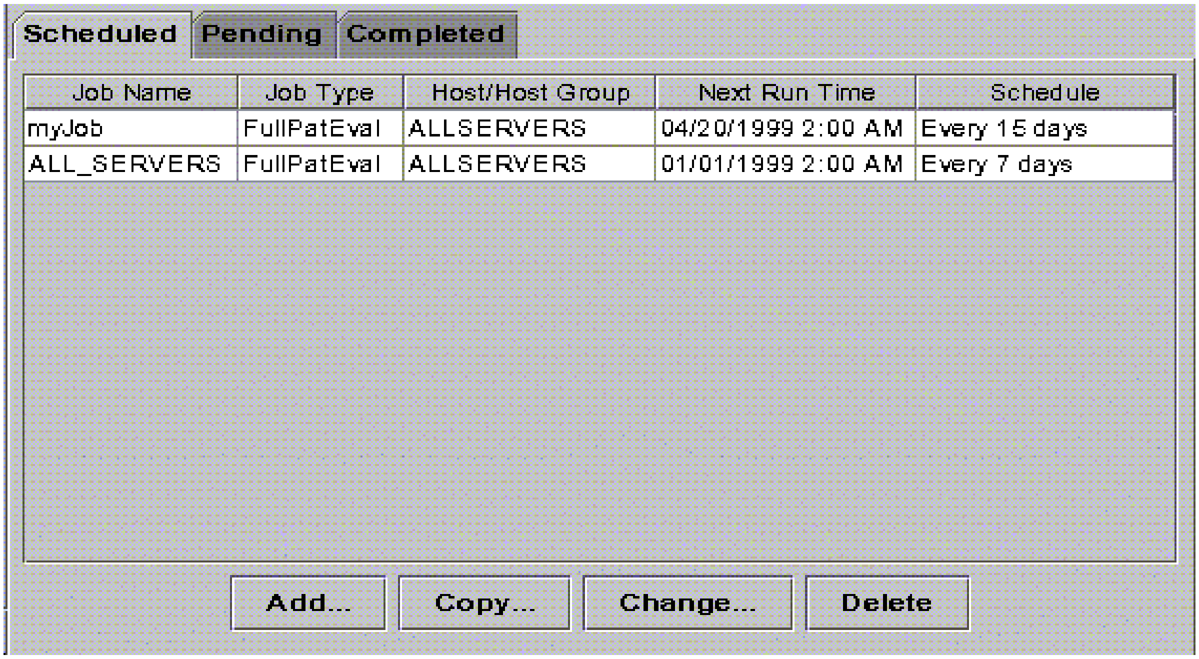
4.1.2.4 Change Host/Host Group Window

When the **Change…** button is pressed and a host has been selected from the host list, the Change Host window is displayed. This window is the same as the Add Host window with the **Host Name** field filled with the existing name.

If the **Change…** button is pressed and a host group has been selected, the Change Host Group window is displayed. This window is the same as the Add Host Group window with the current members of the host group displayed in the **Host Group Members** list for modification. The name of the host group can also be changed.

4.1.3 Scheduled Folder

The **Scheduled** folder displays a list of all the jobs that have been scheduled but not yet run. All jobs start out in this folder unless the job is scheduled to run ASAP (as soon as possible). ASAP jobs are displayed initially in the **Pending** folder.



Job Name	Job Type	Host/Host Group	Next Run Time	Schedule
myJob	FullPatEval	ALLSERVERS	04/20/1999 2:00 AM	Every 15 days
ALL_SERVERS	FullPatEval	ALLSERVERS	01/01/1999 2:00 AM	Every 7 days

Job Name Unique name associated with a job.

Job Type The type of job scheduled.

Host/Host Group Host or Host Group the job will be performed on.

Next Run Time The date and time when the job will be run.

Schedule How often the job will be run.

Add… Creates a new job for scheduling. See Section 4.1.3.1, “Add/Copy/Change Window.”

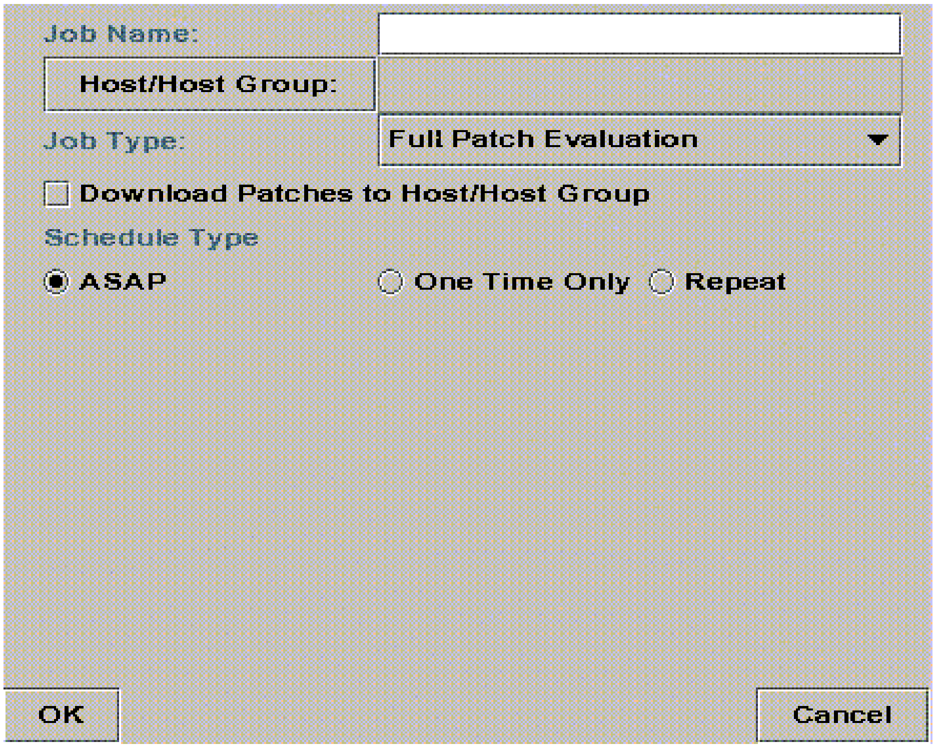
Copy… Copies information from an existing job to create a new job. See Section 4.1.3.1, “Add/Copy/Change Window.”

Change… Modifies information about a scheduled job. See Section 4.1.3.1, “Add/Copy/Change Window.”

Delete Delete the selected job and remove it from the **Scheduled** folder list.

4.1.3.1 Add/Copy/Change Window

Selecting the **Add…**, **Copy…**, or **Change…** buttons will display an Add Job, Copy Job, or Change Job window. These windows are exactly the same except for the name of the window.



Job Name:

Host/Host Group:

Job Type: Full Patch Evaluation

□ Download Patches to Host/Host Group

Schedule Type

● ASAP ○ One Time Only ○ Repeat

Job Name The name of the job being scheduled. This field cannot be changed during a change job operation.

Host/Host Group The name of the host or host group the job will be performed on.

Job Type The type of job to schedule. Currently SafePatch supports only Full Patch Evaluation jobs. A Full Patch Evaluation gathers all patches from the patch database pertaining to the operating system, version, and architecture of the remote system being evaluated.

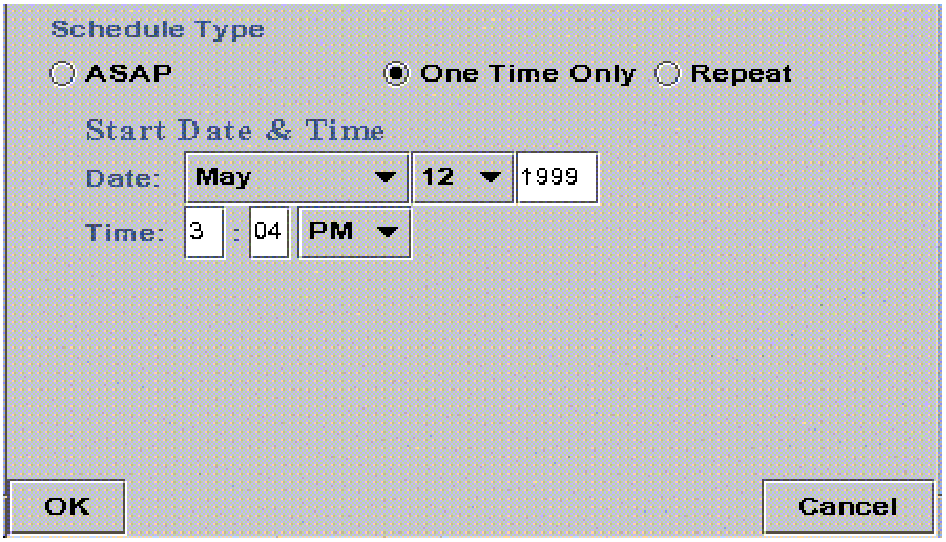
Download Patches to Host/Host Group If selected, SafePatch server will distribute needed patches to the remote system.

Schedule Type Jobs can be run as soon as possible (**ASAP**), once in the future (**One Time Only**), or on a repeated basis (**Repeated)**.

OK Add or change the job and update the list in the **Scheduled** folder. ASAP jobs will be added to the list in the **Pending** folder.

Cancel Returns to the **Scheduled** folder without adding or changing the job.

If the job is scheduled as a **One Time Only**, the bottom of the Add/Change/Copy Job window will look like this:



Schedule Type

○ ASAP ● One Time Only ○ Repeat

Start Date & Time

Date: May 12 1999

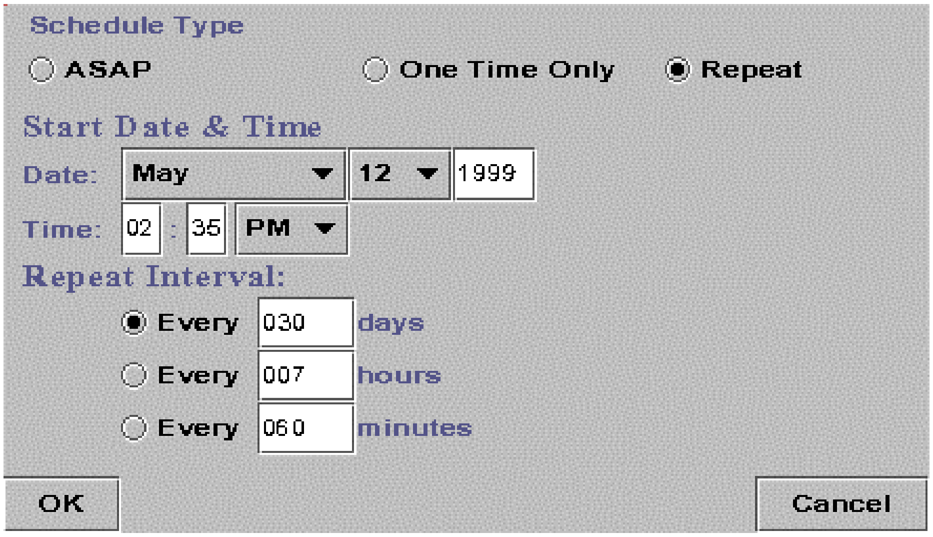
Time: 3 : 04 PM

Start Date & Time The date and time to start processing the job. By default, SafePatch will fill in the current date and time.

Date The starting date in the form of "Month DD YYYY."

Time Start processing at the time defined as "HH:MM", "AM" or "PM."

If the job is scheduled as **Repeat**, the bottom of the Add/Change/Copy Job window will look like this:



Schedule Type

○ ASAP ○ One Time Only ● Repeat

Start Date & Time

Date: May 12 1999

Time: 02 : 35 PM

Repeat Interval:

● Every 030 days

○ Every 007 hours

○ Every 060 minutes

Start Date & Time The date and time to start processing the job. By default, SafePatch will fill in the current date and time.

Date The starting date in the form of "Month DD YYYY."

Time The job will start processing at the time defined as "HH:MM", "AM" or "PM."

Repeat Interval How often the job will be run.

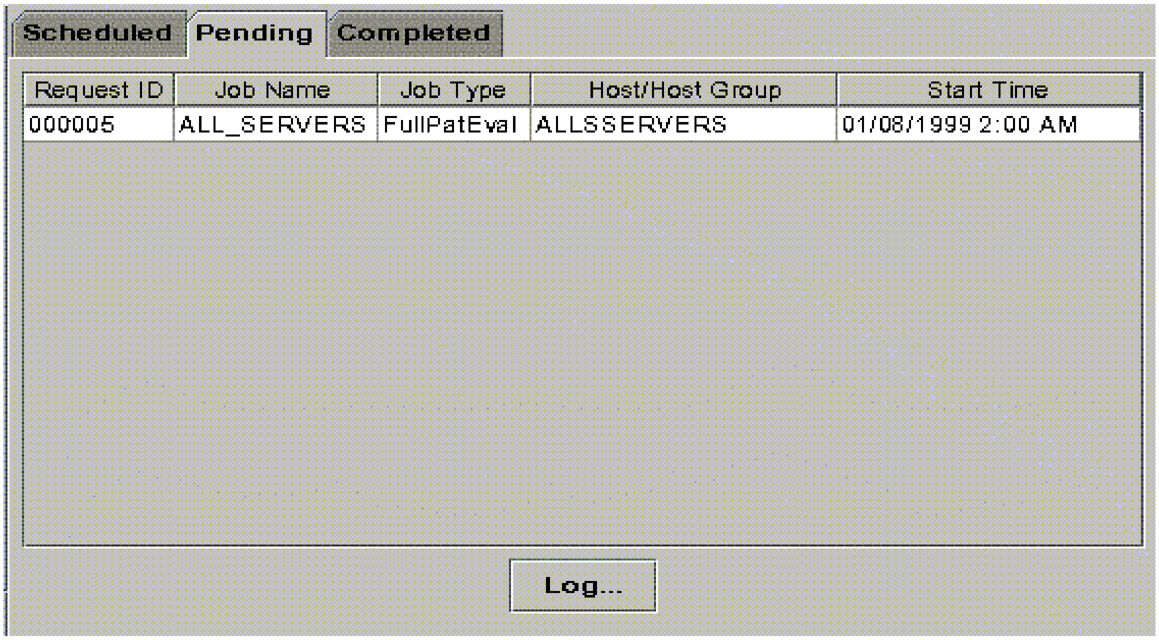
Every ddd days Click on the checkbox and enter how often to repeat the job in days.

Every hhh hours Click on the checkbox and enter how often to repeat the job in hours.

Every mmm minutes Click on the checkbox and enter how often to repeat the job in minutes.

4.1.4 Pending Folder

The **Pending** folder lists all jobs currently being executed. Jobs listed in the **Scheduled** folder will move to the **Pending** folder when the job begins to execute. All jobs created as ASAP jobs will begin their lifecycle in this folder. When a job is moved to the **Pending** folder it will be assigned a unique number called the Request ID. For jobs that are run periodically (Repeat schedule type), a Request ID is assigned to each run of the job. For instance, if a job ALL SERVERS is run every 7 days starting January 1, 1999, then the first time the job runs it will be assigned Request ID 000001 on January 1, 1999. The next time the job runs on January 8, 1999, it may have a Request ID 000005 (if other jobs are running, the Request IDs may not be sequential). This allows each run of a job to be uniquely identified.



Request ID	Job Name	Job Type	Host/Host Group	Start Time
000005	ALL_SERVERS	FullPatEval	ALLSSERVERS	01/08/1999 2:00 AM

Request ID The identification number the Patch Server uses to identify this job.

Job Name The name of the job that has been scheduled.

Job Type The type of job that has been scheduled.

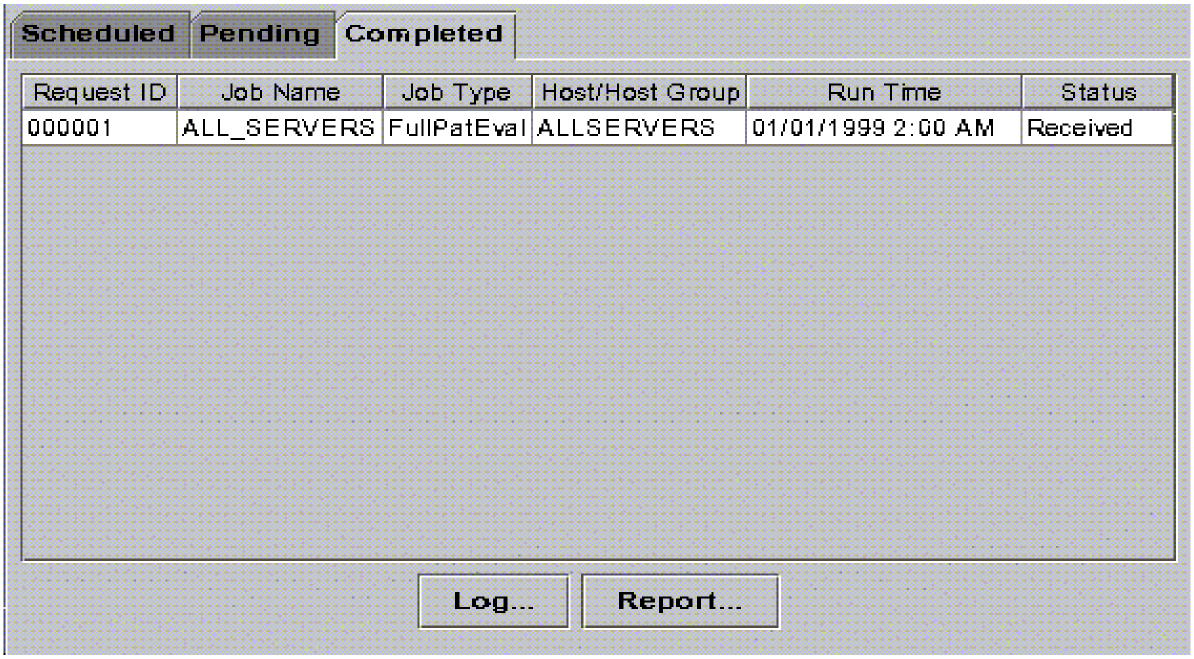
Host/Host Group The name of the host or host group the job is processing the request on.

Start Time Time an evaluation began.

Log… Opens the log file for this job. For more details on viewing the log file, see Section 4.7, “Viewing Log Files.”

4.1.5 Completed Folder

Jobs are moved to the **Completed** folder when the evaluation of *at least* one of the hosts associated with a job is done. This means that a job may not be totally finished when it is moved to this final stage of its lifecycle. For example, if a job is being processed on a host group and Server1 completes processing before the other hosts in the group, the job will move to the **Completed** folder. The log file for this job, as well as the report will continue to be updated as the other hosts in the host group complete processing.



Request ID	Job Name	Job Type	Host/Host Group	Run Time	Status
000001	ALL_SERVERS	FullPatEval	ALLSERVERS	01/01/1999 2:00 AM	Received

Request ID The identification number Patch Server uses to identify this job.

Job Name The name of the job that has been scheduled.

Job Type The type of job that has been scheduled.

Host/Host Group The name of the host or host group the job is processing the request on.

Run Time Time an evaluation began.

Status The status of the first host completing execution of the job. A job status of **Received** indicates that the first host completing execution from the job has finished with no errors. A job status of **Fail** indicates the first host completing execution has finished with errors. To determine why a job failed, see Chapter 6, “Interpreting Log Files”, or Chapter 7, “Troubleshooting.”

Log… Displays log file for a specified job.

Report… Displays the report for a specified job.

4.2 How to Configure the Patch Server

The **Host/Host Group List** identifies the remote systems available for job scheduling. This list needs to be defined before jobs can be scheduled. The system running the SafePatch Server is entered into the host list during the SafePatch installation process and cannot be removed from the list.

This section describes how to add a host to the host list. Hosts can be added one at a time in the case where a new system is added to the network. A group of hosts can also be added all at once as described in section 4.2.3, "Adding Hosts to the Host List from a File." This is especially helpful for first time configurations of SafePatch. This section also describes how to remove systems from the host list and change host names.

4.2.1 Adding a Host to the Host List

To add one remote system to the host list:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the **Add one…** button on the Define Host List window. The Add Host window will be displayed.

3. Enter the **Host Name** of the remote system to be added to the host list. The Patch Server will check the remote system to ensure that it is alive and the SafePatch Agent software is running on the remote system. If the verification fails, a warning message will be displayed.

   Select **Yes** to add the remote system to the host list and return to the Define Host List window.

   Select **No** to return to the Add Host window without adding this host to the host list.

4.2.2 Adding a Host Group

A host group is a conceptual grouping of remote systems. A job can be performed on a host group permitting an easy way to schedule an evaluation on multiple hosts simultaneously. To create a host group:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the **Add Group…** button on the Define Host List window. The Add Host Group window will be displayed.

3. Enter the name of the new host group in the **Host Group Name** field. The host group name must be capitalized. Capitalization of host group names help to quickly distinguish host groups from host names.

4. Select one or more hosts from the **Hosts** list and select the **Add >>** button to move the host(s) to the **Host Group Members** list. The hosts listed in the **Host Group Member** list are the hosts in the new host group. To remove members from the host group, select one or more hosts from the **Host Group Members** list and select the **<< Remove** button. This will move the hosts back to the **Hosts** list.

5. Select the **OK** button to save the new host group. The host group will be listed in the Define Host List window. Select the **Cancel** button to return to the Define Host List window without saving the host group.

4.2.3 Adding Hosts to the Host List from a File

Multiple remote systems can be added to the host list by creating a file listing the names of the remote systems, then using the **Add from file…** option. Each system name should appear on a separate line in the file as shown below. System names cannot contain spaces.

```
Server1
Server2
Server3
ServerN
```

The **Add from file…** option will also create a host group containing all of the systems listed in the file.

To add multiple systems to the host list:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the **Add from file…** button on the Define Host List window. The Add From File window will be displayed.

3. Select the directory where the file resides by:

   a. Select folders in the scroll area to move down the directory tree.
   b. Select a directory from the **Look in** pull down to move up the directory tree.

4. When the file is showing in the scroll window, select it. The file name will be displayed in the **File name** field.

5. Select the **Open** button to read the file and add the hosts to the host list.

   The Patch Server will check the remote systems to ensure that they are alive and the SafePatch Agent is running on these systems. If verification fails, a warning message will be displayed. Select **Yes** to add the host. Select **No** to omit the host. At the end of this verification process, a window will be displayed asking if a host group should be made.

6. Enter the name of the host group and select **OK** to make a host group containing all the systems in the file. Select **Cancel** to return to the Define Host List window without creating a host group.

   The host list in the Define Host List window will be updated to include all hosts listed in the file and (optionally) the host group.

4.2.4 Changing a Host

Occasionally the name of a system may change. To change the system name within SafePatch:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the host to change from the host list.

3. Select the **Change…** button on the Define Host List window. The Change Host window will be displayed with the existing name of the host in the **Host Name** field.

4. Enter the new host name and select the **OK** button to change the name of the host. The old host name will be replaced with the new name in the host list and host groups where it appeared. Select **Cancel** to keep the old host name and return to the Define Host List window.

4.2.5 Changing a Host Group

The members of a host group can be changed by adding new hosts or removing existing hosts from the host group list. To change the membership of a host group:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the host group to change from the host list. Host groups are listed with their names completely capitalized.

3. Select the **Change…** button on the Define Host List window. The Change Host Group window will be displayed.

4. Add and remove host(s) from the host group.

   Select one or more hosts from the **Hosts** list and select the **Add >>** button to move the host(s) to the **Host Group Members** list. The hosts listed in the **Host Group Member** list are the members of the host group. To remove members from the host group, select one or more hosts from the **Host Group Members** list and select the << **Remove** button. This will move the hosts back to the **Hosts** list.

5. Select the **OK** button to save the changes. Select the **Cancel** button to return to the Define Host List window without saving the changes to the host group.

4.2.6 Deleting a Host or Host Group

A host or host group can not be deleted if there are jobs scheduled for this host or host group. Delete the jobs first, then delete the host or host group. Removing a host from the host list will also remove it from any host groups.

To remove a host or host group:

1. Select **Define Host List…** from the **Setup** menu in the Patch Server main window. The Define Host List window will be displayed.

2. Select the host or host group name from the host list.

3. Select the **Delete** button on the Define Host List window. A message will be displayed asking for verification to remove the specified host or host group from the host list.

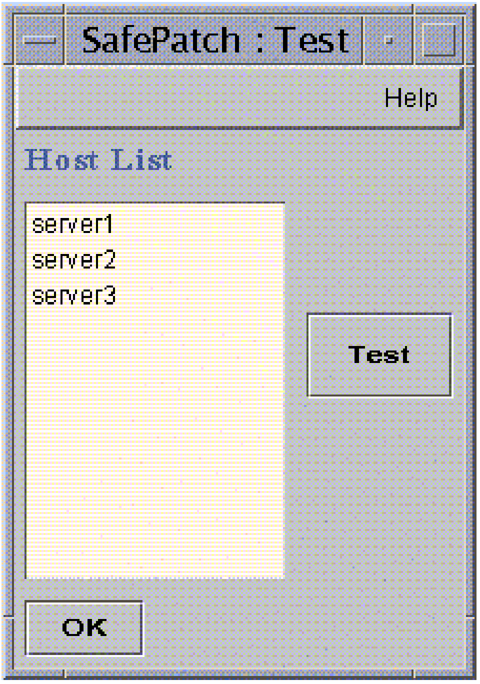
4. Select the **Yes** button to remove the selected host/host group from the host list. Select the **No** button to keep the selected host/host group.

4.3 Testing Communications between the Patch Server and Remote Systems

In order for a job to be performed, the remote system must be running, connected to the network, and running the SafePatch Agent software. Remote systems can be tested to ensure that the SafePatch Agent software is running from the Test Communications window.

To test if a remote system is running the SafePatch Agent software:

1. Select **Test Host Communications…** from the **Control** menu on the main Patch Server window. The Test window will be displayed.



2. Select a host from the **Host List.**

3. Select the **Test** button. A window will be displayed reporting the results of the test. If the Patch Server could not communicate with the host, make sure the host is alive and the SafePatch Agent software is installed and running (see section 7.3).

4. Select the **OK** button to return to the main Patch Server window.

4.4 How to Schedule Jobs

This section provides step-by-step instructions on how to schedule jobs on hosts and host groups.

4.4.1 Scheduling a New Job

To schedule a job:

1. Select the **Add Job…** button on the **Scheduled** folder of the Patch Server window. The Add Job window will be displayed.

2. Enter a unique job name in the **Job Name** field.

3. Select the **Host/Host Group** button to display the host list. Select the host or host group to run the job on. Select the **OK** button in the Host List window. The host name will be displayed in the **Host/Host Group** field on the Add Job window.

4. Select the type of job to run from the **Job Type** pull down list. Currently only Full Patch Evaluation jobs can be run on SafePatch.

5. Check the **Download Patches to Host/Host Group** box if needed patches are to be downloaded to the remote system after the evaluation is complete.

   NOTE: If you are downloading patches on the SafePatch Server and using a download directory other than /tmp, you need to periodically remove the temporary tar files from /temp.

6. Determine how often a job should be run. Jobs can be run once right away (ASAP), once at a specific time (One time only), or repeatedly (Repeat).

   For an ASAP job:
   a. Select the **ASAP** button.

   For a One Time Only job:
   a. Select the **One Time Only** button.
   b. Enter the date to start running the job in the format “Month DD YYYY.” By default the current date will be displayed.
   c. Enter the time in the form “HH : MM” with “AM or PM” selected, when the job should begin execution.

   For a Repeat job:
   a. Select the **Repeat** button.
   b. Specify the time in the form “HH : MM” with “AM or PM” selected, when the job should begin execution.
   c. Fill in the information on how often to run the job. For example, to run the job once a week, select the **Every ddd days** button in the **Repeat Interval** section. Enter 07 into the textfield. To run the job every 12 hours, select the **Every hhh hours** button in the **Repeat Interval** section and enter 12 into the textfield.

4. Select the **OK** button to schedule the job. The job will be added to the list in the **Scheduled** folder or the **Pending** folder (for ASAP jobs only). Select the **Cancel** button to return to the main Patch Server window without scheduling a job.

4.4.2 Copying a Job

One way to schedule a job is to find a job with similar job parameters (*e.g.*, when it starts, what host or host groups it performs on), then copy the job and give it a different name.

To schedule a job by copying an existing job:

1. Select a job to copy from the list of jobs in the **Scheduled** folder of the Patch Server.

2. Select the **Copy…** button. The Copy Job window will be displayed. The Copy Job window is the same as the Add Job window. All the fields except the **Job Name** field will be filled with information from the copied job. The **Job Name** field is left blank.

3. Enter a new **Job Name**.

4. Change any of the job parameters in the Copy Job window.

5. Select the **OK** button to save the new job, and add the job to the list of **Scheduled** jobs. Select the **Cancel** button to return to the main Patch Server window without saving this job.

4.4.3 Changing a Job

To change when a job is scheduled to run and on what hosts or host groups the job is performed:

1. Select a job to change from the list of jobs in the **Scheduled** folder of the Patch Server.

2. Select the **Change…** button. The Change Job window will be displayed.

3. Modify any of the data (except the **Job Name**) in the Change Job window.

4. Select the **OK** button to save the modified job, and update the job information in the **Scheduled** job list. Select the **Cancel** button to return to the main Patch Server window without saving the changes to the job.

4.4.4 Deleting a Scheduled Job

To delete a scheduled job:

1. Select a job from the list of jobs in the **Scheduled** folder of the Patch Server.

2. Select the **Delete** button. A Confirm Delete window will be displayed.

3. Select the **Yes** button to delete the job. Select the **No** button to return to the Patch Server window without deleting the job.

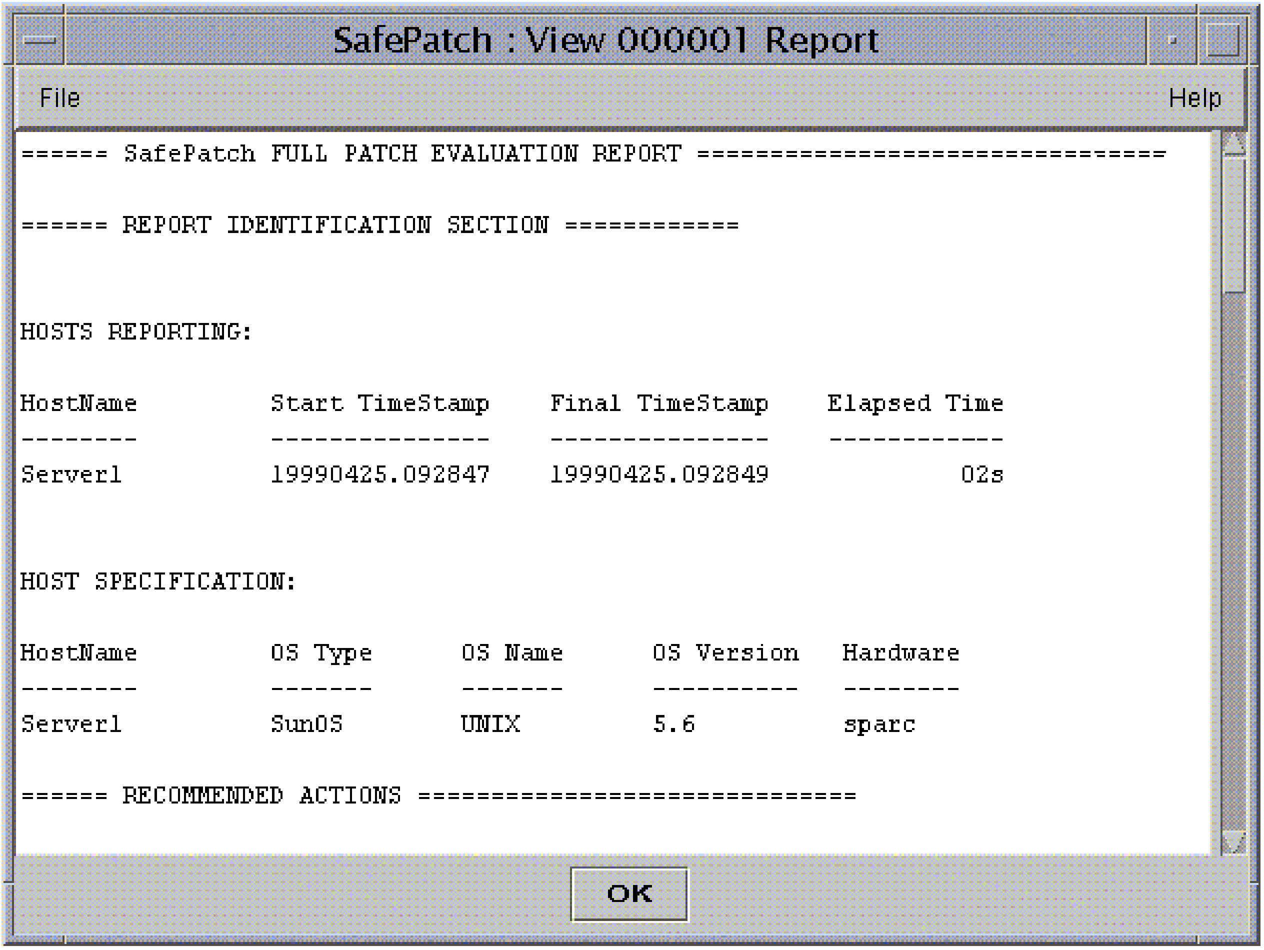
4.5 Viewing a Report

Jobs are moved to the **Completed** folder when the evaluation of *at least one of* the hosts associated with a job is done. This means that a job may not be totally finished when it is moved to this final stage of its lifecycle. If the job is evaluating a host group, the report for this job will continue to be updated as the other hosts in the host group complete processing. The only way to see these new changes is to reselect the **Report…** button periodically.

To generate and view a report for a job:

1. Select a job from the **Completed** folder in the main Patch Server window. Chapter 5, “Interpreting Reports,” provides a detailed description of the contents of a report.

2. Select the **Report…** button. The report viewer will display the report of the selected job.



```
====== SafePatch FULL PATCH EVALUATION REPORT ================================

====== REPORT IDENTIFICATION SECTION ============


HOSTS REPORTING:

HostName         Start TimeStamp    Final TimeStamp    Elapsed Time
--------         ---------------    ---------------    ------------
Server1          19990425.092847    19990425.092849             02s


HOST SPECIFICATION:

HostName         OS Type      OS Name      OS Version   Hardware
--------         -------      -------      ----------   --------
Server1          SunOS        UNIX         5.6          sparc

====== RECOMMENDED ACTIONS ==============================
```

3. When finished viewing the report, select the **OK** button.

4.6 Saving a Report

Reports can be saved to file for archiving. Management of these saved reports is outside of the scope of SafePatch. To save the current report:

1. From the **File** menu on the Report window, select the **Save…** menu item. The Save Report window will be displayed.

2. Select the directory where the report will be saved. Select a folder in the scrolling window to move down the directory tree. Select a directory from the **Look in** pull down menu to move up the directory tree.

3. Enter a name in the **File name** field for a new report file.

   -or-

   To overwrite an existing report, select an existing file from the scroll window. This filename will be displayed in the **File name** field.

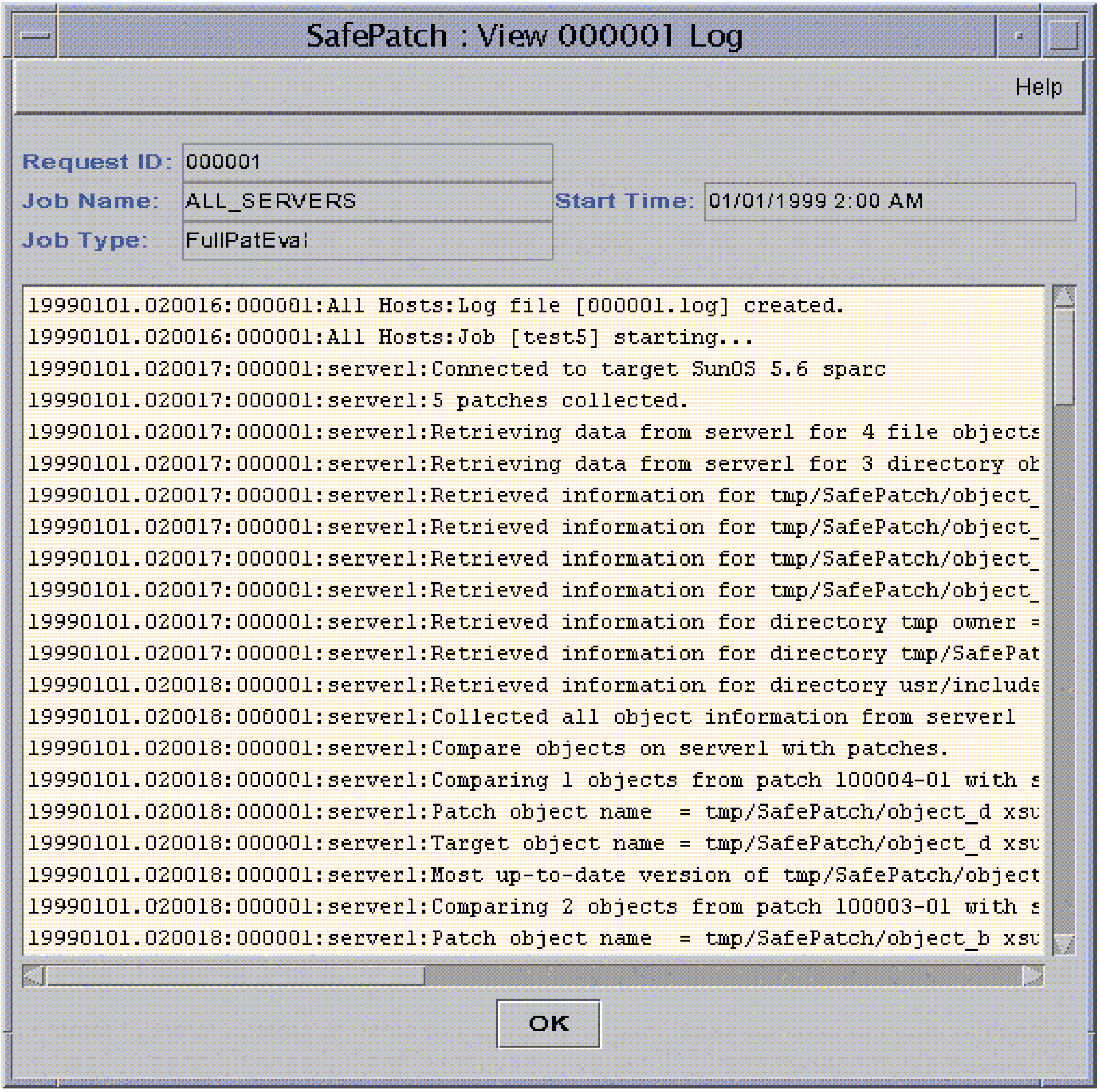
4. Select the **Save** button to save the report to the specified file. Select the **Cancel** button to not save the report and return to the Report window.

4.7 Viewing Log Files

Log files contain messages tracking the job through its lifecycle. For details on how to interpret the log file see Chapter 6, “Interpreting Log Files”. To view a jobs log file:

1. From the **Pending** or **Completed** folder in the main Patch Server window, select a job.

2. Select the **Log…** button. The log viewer will display the log file of the selected job.



Request ID: 000001
Job Name: ALL_SERVERS
Start Time: 01/01/1999 2:00 AM
Job Type: FullPatEval

```
19990101.020016:000001:All Hosts:Log file [000001.log] created.
19990101.020016:000001:All Hosts:Job [test5] starting...
19990101.020017:000001:server1:Connected to target SunOS 5.6 sparc
19990101.020017:000001:server1:5 patches collected.
19990101.020017:000001:server1:Retrieving data from server1 for 4 file objects
19990101.020017:000001:server1:Retrieving data from server1 for 3 directory ob
19990101.020017:000001:server1:Retrieved information for tmp/SafePatch/object_
19990101.020017:000001:server1:Retrieved information for tmp/SafePatch/object_
19990101.020017:000001:server1:Retrieved information for tmp/SafePatch/object_
19990101.020017:000001:server1:Retrieved information for tmp/SafePatch/object_
19990101.020017:000001:server1:Retrieved information for directory tmp owner =
19990101.020017:000001:server1:Retrieved information for directory tmp/SafePat
19990101.020018:000001:server1:Retrieved information for directory usr/include
19990101.020018:000001:server1:Collected all object information from server1
19990101.020018:000001:server1:Compare objects on server1 with patches.
19990101.020018:000001:server1:Comparing 1 objects from patch 100004-01 with s
19990101.020018:000001:server1:Patch object name  = tmp/SafePatch/object_d xsu
19990101.020018:000001:server1:Target object name = tmp/SafePatch/object_d xsu
19990101.020018:000001:server1:Most up-to-date version of tmp/SafePatch/object
19990101.020018:000001:server1:Comparing 2 objects from patch 100003-01 with s
19990101.020018:000001:server1:Patch object name  = tmp/SafePatch/object_b xsu
```

3. When finished viewing the log, select the **OK** button.

5.0 Interpreting Reports

A report is generated for each job. The report contains information for each host the job was performed on. In the case of a job to be performed on a host group, the report will contain information on all hosts in the host group. A report is available when the first host returns its results. The report is continually updated every time a host completes its evaluation. A report is divided into seven sections:

- Report Identification,
- Recommended Actions,
- Current Patches Available,
- Patches Needed, Installed, and Superceded
- Processing Errors,
- File Authentication Results,
- SafePatch Evaluation Summary.

This chapter provides a detailed description of each section in a report.

5.1 Report Identification

The Report Identification section lists the hosts evaluated by this job. If the job evaluated a single host, then only one host will be listed. However, if the job evaluated a host group, then the hosts that have completed the evaluations will be listed in this section. The report will be updated each time an evaluation of a host is completed.

```
======= REPORT IDENTIFICATION SECTION ==========
HOST REPORTING:
HostName        Start TimeStamp         Final TimeStamp          Elapsed Time
Server1         19970923.144000         19970923.150000               20m 00s
Server2         19970923.144000         19970923.150500               25m 00s
Server3         19970923.144000         19970923.151000               30m 00s

HOST SPECIFICATION:
HostName         OS Type       OS Name      OS Version            Hardware
Server1          UNIX          SunOS        5.5                   sparc
Server2          UNIX          SunOS        5.5.1                 sparc
Server3          UNIX          SunOS        5.6                   sparc
```

Host Name Name of the host evaluated by this job.

Start TimeStamp Time that the evaluation started. The time is in the format YYYYMMDD.HHMMSS.

Final TimeStamp Time that the evaluation completed. The time is in the format YYYYMMDD.HHMMSS.

Elapsed Time Amount of time it took to do the evaluation in the form of XXd YYm ZZs (d = days, m = minutes, s = seconds).

OS Type Operating system type (*e.g.,* UNIX).

OS Name Operating system name (*e.g.,* SunOS, Solaris).

OS Version Version of the operating system.

Hardware Hardware type (*e.g.*, sparc or 386).

5.2 Recommended Actions

The Recommended Actions section can have two parts. The first part lists the patches that need to be installed on a host. The patches are listed in the order that they should be installed to avoid conflicts and patch dependencies. In the case of an evaluation on a host group, the patches will be sorted by host name.

Installed patches are checked to see if they have been installed correctly. A patch manipulates (*e.g.*, creates, modifies, or deletes) a set of directories and files. A patch is installed if the checksums of the files in the patch match those on the system. An installed patch is installed correctly if the access control and ownership settings on the files and directories match the recommended settings specified in the patch. The second part of the Recommended Actions sections lists the files and directories that have settings different than the installed patch.

```
======= RECOMMENDED ACTIONS =======================

    For each host listed, the following patches need to be
    installed in the order they appear (typically oldest to
    newest).

    NOTE: Because some older patches may collide with patches that
    were subsequently installed, some of the patches listed below
    will need to be installed even though they may be listed below
    as INSTALLED or SUPERSEDED.

HostName     Patch ID     Patch Date     Description
Server1      102850-04    19971007       OpenWindows 3.5: ff.core
                                         security patch
Server1      105533-01    19971124       SunOS 5.5-syslogd patch
Server1      103399-02    19971224       SunOS 5.5 – XFN source
                                         Modifications for BIND 4.9.3
SETTING CHANGES:
HostName    Filename        Setting    Suggested            Current
Server1     /usr/lib        OWNER      root                 sys
Server1     /usr/lib/nfs/   ACL        bin.user.+r          bin.user.+r
             statd                     -w+x; bin.group.     +w+x; bin.group
                                       +r-w+x; .world.      .+r+w+x; .world
                                       +r-w+x;              .+r-w+x;
```

Host Name: Name of the host needing a patch installed or changes to the owner or permissions settings of a file.

Patch ID: Vendor’s identification of a patch.

Patch Date: Release date of patch.

Description: Short description of patch.

Filename: Name of the directory or file with settings that are different from the patch’s recommended settings.

Setting The file or directory settings that are different on the system than the patches recommended settings. Settings include ownership and permissions. OWNER indicates a difference in the owner of the file/directory. ACL indicates a difference in the group or permissions. The first line under SETTING CHANGES in the report above is an example of a directory (/usr/lib) with the wrong owner. The second line demonstrates a file (/usr/lib/nfs/statd) with the wrong permissions.

Suggested OWNER or ACL setting recommended in the installed patch. In the report above the directory /usr/lib is owned by sys. However, patch 103468-02 recommends that the directory should be owned by root.

ACL settings are in the format:

id.type.perm

In UNIX id is the user or group name; type is user, group, or world (Note: there is no id for world). Perm is a series of characters, each preceded by a "+" to indicate that the permission is granted or "-" to indicate if the permission is denied. Permission characters are operating system dependent. For UNIX permissions include r, w, x, s, S, t, T.

An example of a UNIX file with permissions rw-r-xr-- (0654) in this format is:

bin.user.+r+w-x;bin.group.+r-w+x; .world.+r-w-x

In the report above, owner is bin, group is bin, and permissions are rwxrwxr-x (0775) for the file /usr/lib/nfs/statd. Patch 103468-02 suggests that the permissions on this file should be r-xr-x-r-x (0555).

Current Current OWNER or ACL settings on the host. The format is the same as the Suggested field described above.

5.3 Current Patches Available

The Current Patches Available section lists all the patches collected from the patch database for the evaluation of a host. In this case, Server1 is a sparc machine running Sun OS 5.5 (or Solaris 2.5). All patches for Sun sparc machines running SunOS 5.5 are collected for the evaluation. The Current Patches Available section lists these patches in Patch ID order. As new patches are added to the patch database (see Chapter 3, "Vendor Server") the list of available patches can change from one job to the next.

```
======= CURRENT PATCHES AVAILABLE ==================
OS Type           Patch ID      Patch Date     Description
Server1         102850-03   19960712           OpenWindows 3.5: ff.core
                                                 security patch Server1     Server1
102850-04   19960712             OpenWindows 3.5: ff.core
                                                 security patch
Server1           103468-01     19960517       SunOS 5.5: statd fixes
Server1           103468-02     19961213       SunOS 5.5: statd fixes
Server1           104747-01     19970328       Solstice Firewall-1 2.1c
                                               FD leak in fwd (non_VPN)
Server1           105533-01     11971124       SunOS 5.5: tcp patch
```

5.4 Needed, Installed, and Superseded Patches

The evaluation of a system categorizes patches as NEEDED, INSTALLED and SUPERSEDED. Needed patches are patches that are not installed on the system. However, not all patches marked NEEDED need to be installed. How does this differ from the patches listed in recommended actions? A patch can have one or more revisions. In this case, only the latest revision should be installed. In the example below, patch 102850-04 is listed in the Recommended Actions section above while patch 102850-03 is needed but not recommended for installation.

Installed patches are patches that are currently installed on the host. These patches are listed in the recommended actions section if a needed patch collides with an installed patch. Two patches collide when they have at least one (but not all) file in common and the older patch needs to be installed while the most recent patch is installed. For example, patch 5 released on January 12, 1997, replaces the file /bin/ls. Patch 12 released April 1, 1997, (after patch 5) also replaces file /bin/ls. Patch 5 is not installed while patch 12 is installed on a system. An evaluation of the system determines that patch 5 should be installed. Installing patch 5 overwrites /bin/ls. Therefore, patch 12 must be reinstalled to ensure that the latest version of /bin/ls is installed on the system.

The Patches Superseded section lists patches that no longer apply to a host. For example patch 103468-01 below is superseded because the latest revision of this patch is 103468-02 and is installed on Server1.

```
======= PATCHES NEEDED =============================
Host Name           OS Type         Patch ID        Patch Date
Server1             SunOS 5.5       102850-04       19971007
Server1             SunOS 5.5       102850-03       19960712
Server1             SunOS 5.5       105533-01       19971124
Server1             SunOS 5.5       103399-02       19971224

======= PATCHES INSTALLED ==========================
Host Name           OS Type         Patch ID        Patch Date
Server1             SunOS 5.5       103468-02       19960517

======= PATCHES SUPERSEDED =========================
Host Name           OS Type         Patch ID        Patch Date
Server1             SunOS 5.5       103468-01       19960517
```

5.5 Processing Errors

A patch is listed in this section if SafePatch cannot find or read a file on the remote system. The SafePatch evaluation process compares checksums of files to determine the status of a patch. If the file cannot be checksumed because the file could not be found or read, then the patch status cannot be determined. The comment column in this section indicates if a patch was not evaluated due to missing files or files that cannot be read. If a file cannot be read, the comment will read “Can’t xsum object(s).” Check the permissions on the file and access controls on the SafePatch Agent. The SafePatch Agent may need more permissions in order to read all files it needs.

If files cannot be found then the patch may:

1. Add new files to the system.
2. Include multiple files but recommend replacing only one of these files on a system. These are typically kernel patches. A Sun kernel patch will provide the sun4d, sun4c, and sun4m kernel files but install only the files for the systems hardware.
3. Patch system packages that are not installed on the system (*e.g.,* sendmail, framebuffer graphics).

This version of SafePatch cannot process these types of patches. Before installing any of the patches listed in the Recommended Actions sections, the patches listed in the Processing Errors sections should be reviewed to determine if they should be applied to the system.

```
======= PROCESSING ERRORS (patches not evaluated) ===

     The following patches could not be completely processed due to
     the inability to locate or open related file objects.

Host Name     OS Type      Patch ID     Patch Date     Comment
Server1       SunOS 5.5    104747-01    19970328       Can't xsum object(s).
```

5.6 File Authentication Results

The File Authentication Results section lists each file and directory referenced in the patches collected from the patch database.

```
======= FILE AUTHENTICATION RESULTS ================

     AUTH: Object is Authentic, matches known vendor checksum.
     CUR:  Object is Current, matches up-to-date patch value.

Host Name     Filename                    Auth   Cur   Comment
Server1       usr                         Yes          Patch 103468-02
Server1       usr/lib                     Yes          Patch 103468-02
Server1       usr/lib/nfs                 Yes          Patch 103468-02
Server1       usr/lib/nfs/statd           Yes    Yes   Patch 103468-02
Server1       opt/SUNWfw/bin/fwd          No     No    Can't retrieve xsum
                                                        from Server1
```

Filename: File or directory referenced by at least one of the patches listed in the Current Patches Available section of the report.

Auth: (Short for Authentic) For a directory, Yes indicates that the directory exists on the host. No indicates that the directory could not be found on the host.

For a file, Yes indicates that the checksum of the file matches a checksum of a file in one of the patches listed in the Current Patches Available section. For example, the file came from the vendor (either from a patch or with the original distribution of the operating system) and hasn't been tampered with or customized. No indicates that the checksum does not match any file from any patch in the patch database. There are several reasons why this can happen:

1. The file could be customized. The file could be a configuration file which has a different checksum on every system.

2. The file could be trojan.

3. The file could not be found on the host.

In the first two cases, the comment will be "Unknown" meaning the checksum is unknown. These cases will require further investigation to determine the exact status of a file. In the last case, the comment will be "Can't retrieve xsum from ...."

Cur Yes if the most recent version of this file is installed on the host. No if a patch will update this file.

Comment If the file is not authentic, then this field will include a comment describing the reason why the patch is not authentic, such as “Can’t retrieve xsum...” or “Unknown.” If the file is authentic, then this field will list the patch that it came from or it will be blank if the file was part of the original distribution of the operating system (never patched).

5.7 Summary

The Summary section provides totals of how many patches were evaluated, needed, installed, superseded, and the number of processing errors. In this section, the total number of needed patches are the total number listed under the Patches Needed section, not the Recommended Actions section.

```
======= SafePatch EVALUATION SUMMARY ================

Host Name        Evaluated    Needed    Installed    Superseded    Errors
Server1          119          14        47           57            11
Server2           75          50        20            5             4
GRAND_TOTALS     194          64        67           63            15
```

Evaluated: The total number of patches from the patch database that have been successfully analyzed. All patches that are needed, installed, and superseded are considered evaluated patches. Evaluated also includes the baseline patch.

Needed: The total number of patches listed in the Patches Needed section.

Installed: The total number of patches listed in the Patches Installed section of this report.

Superseded: Total number of patches listed in the Patches Superseded section of this report.

Errors Total number of patches listed in the Processing Errors section of this report.

GRAND_TOTALS The column totals for Evaluated, Needed, Installed, Superseded, and Errors.

6.0 Interpreting Log Files

A log file is created for each job once the scheduler has successfully started the job. The log file contains messages that trace a job through the evaluation process. Since a job can be setup to evaluate one or more remote systems, the log file contains messages for all remote systems in the specified host group.

Messages are formatted as:

YYYYMMDD.HHMMSS: XXXXXX:hostname:message

where:

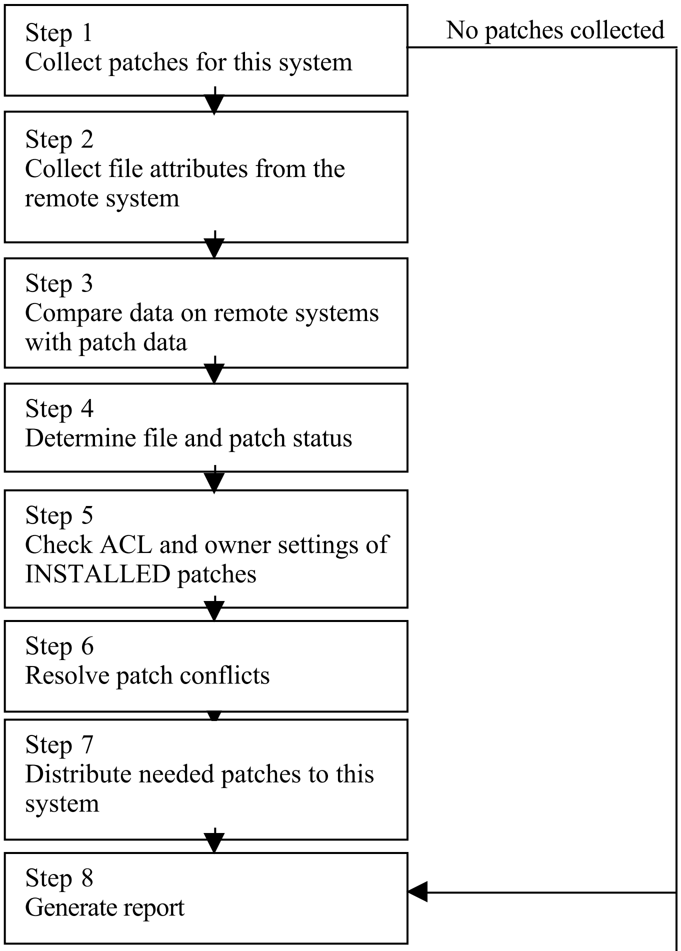
YYYYMMDD.HHMMSS: The date and time that SafePatch wrote the message into the log file. YYYYMMDD is the year, month, and day. HHMMSS is the hour, minute, and seconds represented in military time.

XXXXXX: The request ID which is a unique number assigned to a job.

hostname: Name of the remote system reporting.

message: Status message indicates state of process for evaluating a remote system.

The evaluation process consists of eight general steps. Each of these steps are described in more detail and include sample log messages. The date, time, and request ID are omitted from the sample messages for clarity. Shown below is a flow chart of the evaluation process:


Step 1
Collect patches for this system
No patches collected
Step 2
Collect file attributes from the remote system
Step 3
Compare data on remote systems with patch data
Step 4
Determine file and patch status
Step 5
Check ACL and owner settings of INSTALLED patches
Step 6
Resolve patch conflicts
Step 7
Distribute needed patches to this system
Step 8
Generate report

Step 1. The SafePatch Server connects to the remote system to determine its operating system type (*e.g.*, Solaris, SunOS), version, and architecture. All the patches for this operating system type, version, and architecture are collected from the patch database. The number of patches collected is printed in the log file. In the example below, 129 patches were found in the patch database for a SunOS 5.5.1 (or Solaris 2.5.1) sparc system.

```
All Hosts: Log file [000002.log] created.
All Hosts: Job [Server_Eval] starting . . .
Server1: Connected to target SunOS 5.5.1 sparc
Server1: 129 patches collected.
```

If the number of patches collected is zero, then steps 2 through 6 of the evaluation process are skipped. The number of patches collected can be zero if the patch database is empty or there are no patches for this operating system, version, and architecture in the patch database.

Step 2. The SafePatch Server requests information about files and directories referenced in the collected patches. The remote system returns the checksum, access control list (file permissions), and owner of a file. For a directory the access control list and owner are returned to the SafePatch Server.

In the example below, 196 files and 56 directories are referenced in the collected patches. /usr/sbin/rpc.nisd, /usr/sbin/static, /usr/dt/bin/dtlogin, and /usr/bin/ps are files and directories referenced in a collected patch.

```
Server1: Retrieving data from Server1 for 196 file objects.
Server1: Retrieving data from Server1 for 56 directory objects.
Server1: Retrieving information for usr/sbin/rpc.nisd ( xsum =
c6c4de4a9a3b568c26348621fc5ee23a owner = bin ACL = “bin”.user.+r-w+x,”bin”.group.+r-
w+x,””.world.+r-w+x )
Server1: Retrieving information for usr/sbin/static owner = root ACL =
“root”.user.+r+w+x,”bin”.group.+r+w+x,””.world.+r-w+x
…
Server1: Can’t retrieve information on usr/dt/bin/dtlogin
…
Server1: Retrieving information for usr/bin/ps ( xsum = ERROR - Can’t open file /usr/bin/ps
owner = bin ACL = “bin”.user.+r-w+x,”bin”.group.+r-w+x,””.world.+r-w+x )
…
Server1: Collected all object information from Server1
```

Messages starting with “Can’t retrieve information on ...” indicates that a file or directory cannot be found on the remote system. A file or directory will not exist on the remote system if it belongs to a package (*e.g.,* sendmail, nis, ftp) that was never installed on the remote system.

If the permissions on the file are such that the file cannot be read by the SafePatch Agent then the message “ERROR - Can’t open file ...” will be displayed in either the xsum, ACL, or owner fields. The SafePatch Agent may need to run with more privileges in order to retrieve the necessary data.

Step 3. The checksum of each file in a patch is compared with the checksum collected on the file from the remote system. If the checksums match, as in the case of file /usr/lib/nfs/statd in the example below, then the owner and ACL are compared.

```
Server1: Compare objects on Server1 with patches.
Server1: Comparing 2 objects from patch 104654-04 with Server1’s objects.
Server1: Patch object name = usr/lib/autofs/automountd xsum =
d7a746b2d6c541ed3784dcb6daa576ec
Server1: Target object name = usr/lib/autofs/automountd xsum =
056ef9824aa4ef3921f11bcb775691a5
Server1: Most up-to-date version of usr/lib/autofs/automountd is in patch 104654-04
…
Server1: Comparing 1 objects from patch 104166-03 with Server1’s objects.
Server1: Patch object name = usr/lib/nfs/statd xsum = 8825d6bc3f3a8983aaff34e5614a4fc5
Server1: Target object name = usr/lib/nfs/statd xsum = 8825d6bc3f3a8983aaff34e5614a4fc5
Server1: Most up-to-date version of usr/lib/nfs/statd is in patch 104166-03
Server1: Patch’s usr/lib/nfs/statd ACL = “bin”.usr.”r-w+x, “bin”.group.+r-w+x.””.world.+r-w+x
Server1: Target’s usr/lib/nfs/statd ACL = “bin”.usr.”r-w+x, “bin”.group.+r-w+x.””.world.+r-
w+x
Server1: Target’s usr/lib/nfs/statd owner is bin should be bin
```

If “ERROR - Can’t retrieve xsum from ...” is displayed for the target’s xsum the file could not be opened by the SafePatch Agent.

Step 4. Each file in a patch is evaluated to determine its status. Valid file statuses include:

NOT_SET An error occurred in gathering information from the remote system.

UNKNOWN The checksum on the remote system does not match any patch file checksums or an error occurred in gathering information from the remote system.

NEEDED An older file exists on the remote system and it should be replaced.

OBSOLETE A more recent file is installed on the remote system.

INSTALLED The patch’s file is installed on the remote system.

The patch is given a status based on the status of all the files in the patch. Patch status include:

NOT_INSTALLED One or more files have the status NOT_SET due to errors in gathering data from the remote system.

INSTALLED One or more files are INSTALLED and zero or more files are OBSOLETE.

SUPERSEDED All the files have the status OBSOLETE.

NEEDED If one of the above status cannot be applied to a patch, then the patch is NEEDED.

In the example below, patch 104654-04 patches the file /usr/lib/autofs/automountd. The file automountd is NEEDED. Therefore, the patch 104654-04 is needed.

```
Server1: Determining object status.
Server1: Tagging objects in patch 104654-04
Server1: usr/lib/autofs/automountd is NEEDED.
Server1: Patch 104654-04 is NEEDED.
…
Server1: Determined patch status.
```

Step 5. For each INSTALLED patch, determine if the files and directories have the recommended ACL and owner settings.

In the example below, patch 105004-09 recommends directory /kernel has rwr-xr-x permissions and should be owned by root. This recommendation matches the settings on the remote system.

```
Server1: Compare directories on Server1 with patches.
Server1: Comparing 2 directories from patch 105004-09 with Server1’s directories
Server1: Patch directory = kernel
Server1: Target’s directory = kernel
Server1: First reference of directory kernel is in patch 105004-09
Server1: Patch’s kernel ACL = “root”.user.+r+w+x, “sys”.group.+r-w+x,””.world.+r-w+x
Server1: Target’s kernel ACL = “root”.user.+r+w+x, “sys”.group.+r-w+x,””.world.+r-w+x
Server1: Target’s kernel owner = root
Server1: Patch’s kernel owner = root
Server1: Patch directory = kernel/fs
Server1: Target’s directory = kernel/fs
. . .
```

Step 6. Not all NEEDED patches need to be installed. A patch that conflicts with another patch or is obsolete may be given a status of NEEDED in step 4 but should not be installed. Also some patches are revisions of other patches (*e.g.*, 104166-03 is a revision of 104166-02). In this case only the most recent patch needs to be installed. Step 6 does the final checking on NEEDED patches to determine the minimum set of patches that should be installed.

```
Server1: Determining recommended action.
Server1: Add 103591-09 to the needed list.
```

Step 7. Needed patches will be distributed in a tar file to remote systems if the patch distribution option was selected on the Add Job window. A patch distribution key of download indicates that the patches be distributed to the remote system. In this case, three patches are put into the tar file server1.000002.tar and moved to /tmp directory on Server1.

```
Server1: Preparing patches for distribution - patches needed = 3
Server1: Patch distribution key = download
Server1: Created patch distribution file = /tmp/server1.000002.tar
```

Step 8. The final step is to generate a report.

```
Server1: Evaluation completed, generating report.
Server1: Writing results to /safepatch-0.9/binm/D/reqs/ReqResults/R00002Server1
Server1: Job Completed.
Server1: All Hosts: Job [Server_Eval] completed
```

7.0 Troubleshooting

This section describes the most common issues encountered when using SafePatch. Section 7.1, “Patch Collection Failed” details what to do when the Vendor Server is not collecting patches from the vendor’s ftp site. Section 7.2, “Failed Jobs” provides help on jobs that have been moved to the **Completed** folder with a Failed status. Section 7.3, “SafePatch Agent Not Running” describes how to check the SafePatch Agent process running on remote machines. Section 7.4, “Forgotten Password” outlines what to do when the SafePatch password is forgotten. Section 7.5, “Invalid Keys” outline what to do when the keys generated at start-up have been corrupted and need to be replaced.

7.1 Patch Collection Failed

A scheduled patch collection process should generate a *.MINE log file when it has completed collecting patches from a vendor’s ftp site. As shown below, a *.MINE file contains a list of filenames collected from the vendor server’s ftp site.

```
103187-34.tar.Z
104654-04.tar.Z
104655-04.tar.Z
105004-10.tar.Z
105310-04.tar.Z
105375-04.tar.Z
```

A *.SMELT log file is generated if at least one patch is collected from the vendor’s ftp site. Patches collected from a vendor’s ftp site are then converted to the standard patch format. The *.SMELT file lists the patches successfully converted. These files should be saved in the RAWDB directory. A sample output is provided below:

```
Start processing the patches from /safepatch/RAWDB/Sun_FREE.19980216.010000.0.MINE
at Mon Feb 16 01:07:18 PST 1998.
----------------------------------------------------
Total number of patches to be processed:            6

Created: 19980213.01.SMI.103187-34
Created: 19980213.01.SMI.104654-04
Created: 19980213.01.SMI.104655-04
Created: 19980213.01.SMI.105004-10

Patches converted successfully:                     4
----------------------------------------------------

Found patches failed to be converted due to one of the problems listed below:
1. Files IO problems/errors.
2. Unknown permission setup.
3. Problem MD5.
4. Missing info.

104468-10
Patches may need to be converted manually:          1
----------------------------------------------------
Found number of patches already have patch spec created:     1
Found number of non-security patches:                2

Found number of older version patches:               1

End processing the patch list from /safepatch/RAWDB/Sun_FREE.19980216.010000.0.MINE
at Mon Feb 16 01:10:38 PST 1998.
---------------------------------------------------
```

Listed below are some things to look for when problems are encountered with patch collection:

1. If a file vendorName.date.time.0 exists in this directory and no *.MINE or *.SMELT exists, then the owner and permissions on the patch conversion script are wrong. The patch conversion script is the file that is used to convert the patches to the SafePatch standard patch format. The conversion script permissions should also be read/executable for the owner. The script files are located in the safepatch-0.9/binm/ES/smelt_script directory. A sample from the vendorName.date.time.0 file is included below:

```
--12:00:04--  ftp://sunsolve1.sun.com:21/pub/patches
           => `/safepatch/RAWDB/sunsolve1.sun.com/pub/.listing'
Connecting to sunsolve1.sun.com:21... connected!
Logging in as anonymous ... Logged in!
==> TYPE I ... done.  ==> CWD pub ... done.
==> PORT ... done.    ==> LIST     done.
/safepatch/RAWDB/sunsolve1.sun.com/pub/.listing: Permission denied   ←
--12:00:05--  ftp://sunsolve1.sun.com:21/pub/patches
           => `/safepatch/RAWDB/sunsolve1.sun.com/pub/patches.1'
Connecting to sunsolve1.sun.com:21... connected!
Logging in as anonymous ... Logged in!
==> TYPE I ... done.  ==> CWD pub ... done.
==> PORT ... done.    ==> RETR patches ...
No such file `patches'.

FINISHED --12:00:06--
Downloaded: 0 bytes in 0 files
```

It is important to remember that all directories under the SafePatch tree structure have the same owner permissions as the person logging into SafePatch and running the application.

2. If the RAWDB directory is empty, the patch collection process did not run. Check the daemon processes to make sure that they are running. To do this, see Section 3.1.1.2, “Control Menu.”

3. The patches listed after the comment “Found patches failed to be converted…” could not be converted to the standard patch format. The sample log file above lists patch 104468-10 as having failed to be converted. If all the patches collected are listed in this section (i.e., other sections such as “Total number of patches to be processed: 0” don’t list any patches) then check the permissions and ownership of the RAWDB directory. The directory should have rwxr-xr-x (0 755) permissions and be owned by the SafePatch administrator.

4. If the conversion process fails for some but not all patches, send the *.smelt log file and the patches for which the conversion process failed to safepatch@cheetah.llnl.gov.

7.2 Failed Jobs

If a job has been moved to the **Completed** folder with a **Status** of “Failed,” the job did not successfully run. Check daemon processes to make sure they are still running. To do this, see Section 3.1.1.2, “Control Menu.” Also check the remote system to ensure that the system is alive and the SafePatch Agent is still running. See section 7.3, “SafePatch Agent Not Running,” for more information.

7.3 SafePatch Agent Not Running

If the **Test Host Communications** feature was used, and a message that the host is unreachable was displayed, the SafePatch Agent software was not detected on the remote machine. If the SafePatch Agent software was not installed, install it using the safepatch_agent-0.9.tar.Z file (see the installation guide). If the Agent is installed, the SafePatch Agent process needs to be restarted. Log onto the remote system and run the StopClient executable. Restart the SafePatch Agent by running the StartClient executable in the safepatch-agent-0.9/binr/agent in the safepatch-agent-0.9/binr/agent directory.

7.4 Forgotten Password

If the SafePatch password is forgotten, the SafePatch password file can be deleted. This forces SafePatch to prompt for a new password the next time the application is started. The password file, “.PWsig” is located in the safepatch-0.9/binm/D directory. Remove this file, and start the SafePatch Server. A window will be displayed prompting for a new password (see section 2.1, “Logging into the SafePatch Server,” for more details).

7.5 Invalid Keys

The keys generated at start-up have been corrupted. New keys can be generated by:

1. Remove the password file if one exists
   >> rm SafePatch /binm/D/.pwsig

2. Remove keys
   >> rm SafePatch/binm/D/certinfo/crt-host.*
   Where host is the name of the host running the SafePatch Server

3. Restart SafePatch. You will be prompted for a new password.

4. Enter the new password and verify it.

5. Select OK.

6. New keys and a password file will be generated.

Glossary

ACL

Access Control List or read-write-execute settings for owner, group and world in the UNIX environment. ACL settings are in the format:

id.type.perm

where id is the user or group name; type is either user, group, or world (Note: there is no id for world). Perm is a series of characters, each preceded with a “+” or “-“ granting or denying permissions. Permission characters for UNIX include r, w, x, s, S, t, T.

Daemon processes

Two processes must be running at all times for the SafePatch Server to execute. One process controls the scheduling of jobs and patch collection from vendor sites. The other process controls the execution of jobs. These processes are referred to as daemon processes. The daemon processes can be started and stopped from both the Vendor Server and Patch Server.

Full Patch Evaluation

Type of evaluation job that can be scheduled from the Patch Server. Currently this is the only type of job SafePatch supports. A full patch evaluation gathers all patches from the patch database pertaining to the operating system, version, and architecture of the remote system being evaluated. Future versions of SafePatch may have other types of jobs such as a patch or file query (*i.e.*, is this patch installed on a system or is this file installed on a system).

Host

A remote system running the SafePatch Agent software.

Host Group

One or more remote systems grouped together for ease of scheduling jobs on the collection of systems. A unique name is associated with the group.

Job

The evaluation of one or more remote systems. Jobs are scheduled and tracked using the Patch Server.

Object

A file or directory.

Patch Database

The directory storing patches in the SafePatch standard patch format. These patches are used in the evaluation processes. The patch database is also referred to as the PSDB (Patch Spec DataBase).

Patch Server

Part of the SafePatch Server controlling the scheduling and tracking of jobs.

RAWDB

Directory where the patches collected from a vendor’s ftp site are stored. The location of this directory is determined at installation.

Request ID

A unique number given to job. A job scheduled to run once or ASAP will have one request ID associated with it. A job scheduled to run repeatedly will have a request ID associated with each run of this job in order to distinguish each run.

SafePatch Agent

A software package installed on a remote system. The SafePatch Agent is a lightweight process that responds to the commands and requests of the SafePatch Server.

SafePatch Server
A software package installed on a central server from which the analysis of remote systems can be scheduled and monitored. The SafePatch Server is analogous to a backup server. The SafePatch Server is composed of two parts: the Vendor Server and the Patch Server.

Standard Patch Format
A format for patch information that is vendor-independent. All patches collected by SafePatch are converted to this format. A standard format permits SafePatch to operate on any platform

Target
A remote system.

Vendor Server
Part of the SafePatch Server responsible for the monitoring of ftp sites, collecting new and updated patches, and converting patches to a standard patch format.

xsum
Abbreviation for a MD5 or SHA-1 message digest.